

ORACLE® HYPERION SMART VIEW FOR OFFICE, FUSION EDITION

RELEASE 11.1.1



Smart View User's Guide, 11.1.1

Copyright © 2004, 2008, Oracle and/or its affiliates. All rights reserved.

Authors: EPM Information Development Team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable: U.S. GOVERNMENT RIGHTS: Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

Contents

| | Documentation Accessibility |
|--------------|---|
| Chapter 1. I | ntroduction to Smart View |
| | Smart View |
| | Smart View Benefits |
| | Smart View Toolbar |
| | Toolbar Buttons |
| | Shortcut Menus |
| | Login Requirements |
| | Using Google Search with Smart View |
| | For Microsoft Office 2007 Users |
| Chapter 2. I | Features and Data Source Providers |
| | Ad Hoc Features and Supported Data Source Providers |
| | Query Designer Features and Supported Data Source Providers |
| | Dynamic Data Access Features and Supported Data Source Providers |
| | Hyperion Visual Explorer Features and Supported Data Source Providers 24 |
| | Data Form Features and Supported Data Source Providers 24 |
| | Function Features and Supported Data Source Providers |
| | Reporting and Analysis Document Import Features and Supported Data Source |
| | Providers |
| Chapter 3. I | nstalling and Enabling Smart View and Visual Explorer |
| | Installing Smart View |
| | Enabling and Disabling Smart View |
| | Smart View and Essbase Spreadsheet Add-in |
| | Information for Administrators |
| | Running Silent Installations |
| | Checking for New Smart View Versions |
| | Compatibility with Data Sources |
| | Uninstalling Smart View |

| Chapter 4. (| Connecting to Data Sources |
|---------------|--|
| | The Data Source Manager and Connections |
| | Common Provider Connections |
| | Connecting to Common Provider Data Sources |
| | Disconnecting Common Provider Connections |
| | Setting Data Source Manager Views |
| | Associating a Connection to a Worksheet |
| | Adding and Deleting Connections |
| | Setting Provider Properties |
| | Independent Provider Connections |
| | About Independent Provider Connections |
| | Adding a Data Source Connection |
| | Connecting to a Data Source |
| | Associating a Data Source Connection to a Worksheet |
| | Editing a Data Source Connection |
| | Setting a Default Connection |
| | Deleting a Data Source Connection |
| | Disconnecting from a Data Source |
| | Checking for Active Connections |
| | Disconnecting a Worksheet from its Data Source |
| | Migrating Essbase Connections to the Data Source Manager |
| | Simulation and Forecasting Workbooks |
| | Working With Simulation and Forecasting Workbooks |
| | Managing Simulation and Forecasting Workbooks 43 |
| | Changing Passwords |
| Chapter 5. 9 | Smart Slices |
| | Creating Queries from Smart Slices |
| | Smart Slices, Ad Hoc Analysis, and Data Forms |
| | Creating Smart Slices |
| | Setting Smart Slice Data Boundaries |
| | Setting Smart Slice Preferences |
| Observan C. 7 | Flor Donard Declarer |
| Chapter 6. I | The Report Designer |
| | Creating Reports |
| | Report Types |
| | Controlling Reports |
| | Creating a Slider from One Query |
| | Creating a Slider from Joined Queries |
| | Deleting Reports |

| (| Cascading Reports and Ad Hoc Grids | 55 |
|-------------------|---|----|
| | Cascading Smart Slice Reports | 55 |
| | Cascading Ad Hoc Grids | 55 |
| Chapter 7. Workin | ng with Data Forms | 57 |
| 7 | Working with Data Forms in Excel Worksheets | 57 |
| | Opening Data Forms in Excel | 59 |
| | Navigating in Data Forms | 59 |
| | Viewing Data Form Instructions | 60 |
| | Viewing Multiple Data Forms in Excel | 60 |
| | Selecting Members for Financial Management and Hyperion Enterprise Data Forms | 60 |
| | Adding Members to Financial Management Data Forms | 62 |
| | Using Financial Management Linked Forms | |
| | Selecting a Range of Cells | |
| | Saving Custom Formats | 63 |
| | Searching for a Page in Planning | 63 |
| | Copying and Pasting Cells | 64 |
| | Entering Percentage Values in Planning | 64 |
| | Entering Date Values in Planning | 64 |
| | Writing #Missing Values | 65 |
| | Adding Cell Text in Data Forms | 65 |
| | Viewing and Editing Cell Text in Data Forms | 66 |
| | Submitting Cell Text in Planning | 66 |
| | Working with URLs Linked to Cells | 66 |
| | Subtotaling Values in Planning | 67 |
| | Changing Currency in Planning Data Cells | 67 |
| S | Submitting Data | 68 |
| 7 | Working with Business Rules for Planning | 69 |
| | Launching Business Rules in Excel | 69 |
| | Launching Business Rules From a Data Form | 70 |
| | Entering Runtime Prompts for Business Rules | 70 |
| | Executing the Calculate Data Form and Calculate Currencies Business Rules | 72 |
| I | Adjusting and Spreading Data Values | 72 |
| | Adjusting Values | 73 |
| | Spreading Data for Time Periods | 73 |
| | How Spreading Data Works | 74 |
| | Spreading Data with Cell Locking in Planning | 77 |
| | Examples of Spreading Data with Cell Locking | 78 |
| 7 | Working With Supporting Detail | 79 |
| | | |

| | Adding Supporting Detail |
|--------------|---|
| | Totaling When Supporting Detail Cells Are Blank |
| | Working with Supporting Detail Hierarchy 8 |
| | Viewing or Changing Supporting Detail |
| | Synchronizing Supporting Detail with Essbase |
| | Working with Formulas in Data Forms 8 |
| | Working with Smart Lists |
| Chapter 8. Q | Query Designer |
| | Query Designer Features |
| | Query Designer Worksheets |
| | Creating Queries from Default Reports |
| | Extracting Queries from Reports |
| | Editing Queries |
| | Filtering Members in Queries |
| | Filtering Data |
| | Analyzing Time-Related Data in Query Designer9 |
| | Rerunning Queries |
| | MDX Queries |
| Chapter 9. W | Vorking with Ad Hoc Analysis |
| | The Ad Hoc Grid9 |
| | Setting Ad Hoc Options |
| | Selecting Members for Ad Hoc Analysis |
| | Filtering Members 9 |
| | Member Selection Differences |
| | Dragging and Dropping Dimension Members |
| | Working with Dynamic Time Series Members |
| | Formatting Data Cells in the Grid |
| | Refreshing the Grid |
| | Zooming In on Dimension Members |
| | Zooming Out on Dimension Members |
| | Retrieving Attribute Dimensions and Members |
| | Viewing the Qualified Name of a Member |
| | Displaying Aliases for Member Names |
| | Accessing Data from a Hybrid Analysis Relational Source |
| | Pivoting Dimensions |
| | Keeping Data |
| | Removing Data |
| | Adding Cell Text in Ad Hoc Grids |

| | Viewing Cell Text in Ad Hoc Grids |
|---------|---|
| | Calculating Data |
| | Selecting a Calculation Script |
| | Inserting Rows and Columns |
| | Translating Data |
| | Consolidating Data |
| | Submitting Data 114 |
| | Working with Formulas in Ad Hoc Grids |
| | Drill-Through Reports |
| | Data Perspective |
| | Member Perspective |
| Chapter | 10. Using Free-Form Reporting to Retrieve Data |
| | About Free-Form Reporting |
| | Smart View Grid Components |
| | Free-Form Guidelines |
| | Working with Attribute Dimensions |
| | Entering Dynamic Time Series Members |
| | Submitting Dirty Cells |
| | Comment Handling |
| | Preserving Comments, Formulas, and Format |
| | Handling the #Missing and the #No Access Label |
| | Valid and Invalid Grids |
| | Retrieving in Free-Form Mode |
| | Resolving Dimension Names |
| | Retrieving Attribute Dimensions in Free-Form Mode |
| | Resolving Member Names in Free-Form Grids |
| | Answering Business Questions Using Free-Form |
| | Constructing an Example Free-form Report Using a Hyperion Enterprise Data |
| | Source |
| | Analyzing Time-Related Data in Free-Form Mode |
| | Highly Formatted Free-Form Report Example |
| | Creating and Distributing Report Templates Using Free-Form |
| | Creating a Base Grid Using Dimension Name Resolution |
| | Using Free-Form to Expand the Layout |
| | Leveraging Excel Formulas |
| | Formatting the Grid |
| | Using the POV as a Page Selector |
| | Creating Additional Reports |
| | Using the POV as a Page Drop-down |

| | Tips |
|-----------|---|
| | Retrieving Data into Asymmetric Reports |
| Chapter 1 | 1. Accessing Dynamic Data Across Microsoft Office |
| | Copying and Pasting Live Data into Word and PowerPoint |
| | Changing POV in Word and PowerPoint |
| Chapter 1 | 2. Visual Explorer |
| | Viewing Data in Graphical Format |
| | Viewing Data from Excel |
| | Viewing Data from Word or PowerPoint |
| | Starting Visual Explorer Independently of Smart View |
| | Connecting to a Data Source from Visual Explorer |
| Chapter 1 | .3. Importing Reporting and Analysis Documents into Microsoft Office |
| | Importing Reporting and Analysis Documents |
| | Editing and Refreshing Documents |
| | Refreshing Reporting and Analysis Documents |
| | Financial Reporting and Web Analysis Import Formats |
| | Adding Security Certificates for SSL-enabled EPM Workspace Servers |
| | Importing Interactive Reporting Documents |
| | Importing Interactive Reporting Documents into Excel |
| | Importing Interactive Reporting Documents into Word and PowerPoint 170 |
| | Editing Interactive Reporting Documents |
| | Importing Financial Reporting Documents |
| | Importing Financial Reporting Documents into Excel |
| | Importing Financial Reporting Documents into Word and PowerPoint 17. |
| | Editing Financial Reporting Documents |
| | Creating Templates in PowerPoint Documents |
| | Refreshing PowerPoint Templates |
| | Importing Production Reporting Documents |
| | Importing Production Reporting Jobs into Excel |
| | Importing Production Reporting Jobs into Word and PowerPoint |
| | Importing Production Reporting Job Outputs into Excel, Word, and PowerPoint 178 |
| | Editing Production Reporting Jobs |
| | Importing Web Analysis Documents |
| | Importing a Web Analysis Document or Document Objects |
| | Editing Web Analysis Documents |
| | Using Smart Tags to Import Reporting and Analysis Documents |

| Chapter 14. Fu | nctions |
|-----------------|--|
| | About Functions |
| | Examples |
| | The Function Builder |
| | Selecting Members for Functions |
| | HsGetValue |
| | HsSetValue |
| | HsCurrency |
| | HsDescription |
| | HsLabel |
| | HsGetText |
| | HsSetText |
| | Accessing Functions Using Smart Tags |
| | Retrieving Values |
| | Retrieving Cell Text |
| | Retrieving Entity Currency |
| | Displaying the POV Description |
| | Removing Smart Tags for a Single Instance |
| | Stopping Hyperion Smart Tag Recognition |
| | Recognizing Smart View Smart Tags Again |
| | Creating Functions Manually |
| | Editing Functions |
| | Running Functions |
| | Common Function Error Codes |
| Chapter 15. Wo | orking with Offline Planning |
| | About Working Offline |
| | Taking Data Forms Offline |
| | Working with Data Forms Offline |
| | Synchronizing Data to the Planning Server |
| | Refreshing the Offline Data Form Definition and Data |
| | Entering Data with Menus |
| Chapter 16. Use | er Preferences |
| | Ad Hoc Options |
| | Suppressing Types of Data |
| | Indenting Member Names |
| | Navigating Without Data |
| | Double-clicking in Ad Hoc Operations |
| | Undo and Redo |
| | |

| | Zoom Options in Connected Sheets | 210 |
|-----------|---|-----|
| | Member Retention Options | 211 |
| | Ancestor Positions in Hierarchies | 211 |
| | Setting the Provider ServicesConnection URL | 212 |
| | Display Options | 212 |
| | Replacement Labels | 212 |
| | Cell Status | 213 |
| | Member Name Display Options | 214 |
| | UI Colors | 214 |
| | Format Strings | 214 |
| | Scale and Decimal Display Options | 215 |
| | Using Excel Formatting | 215 |
| | Formula Preservation After POV Changes | 216 |
| | Reducing Excel File Size | 217 |
| | Displaying and Logging Messages | 217 |
| | Setting Cell Styles | 218 |
| | Cell Style Precedence | 218 |
| Chapter 1 | 17. Using the POV Manager | 221 |
| | Editing the Point of View | 221 |
| | Selecting Dimension Members for the POV | 222 |
| | Copying and Pasting a POV | 223 |
| | Deleting a POV | 224 |
| | Printing POV Members in Header and Footer | 224 |
| Chapter 1 | 18. Using the Migration Utility | |
| | Considerations Before Converting Workbooks | 227 |
| | Converting One Workbook | 228 |
| | Converting Multiple Workbooks | |
| | Migrating Connections for Functions | 229 |
| Chapter 1 | 19. VBA Functions | 231 |
| | Using VBA Functions for Smart View | |
| | Migrating Legacy VBA Applications | |
| | Creating a Visual Basic Module | |
| | Using Smart View VBA Functions | |
| | Declaring Functions | 232 |
| | Calling Functions | 233 |
| | Dynamic Link Views | 234 |
| | VBA Parameters | 236 |

| VBA Return Values |
|---|
| VBA Local Return Values |
| VBA Functions |
| About Visual Basic Menu Equivalent Functions |
| Visual Basic Menu Functions |
| Error Codes |
| ccessibility |
| Enabling Accessibility for Smart View |
| Keyboard Equivalents |
| Hyperion Menu Keyboard Equivalents |
| Independent Provider Connections Keyboard Equivalents |
| Common Provider Connections Keyboard Equivalents |
| Report Designer Keyboard Equivalents |
| Accessibility in Office 2007 |
| Index |
| |

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

1

Introduction to Smart View

In This Chapter

| Smart View | 15 |
|-------------------------------------|----|
| Smart View Benefits | 15 |
| Smart View Toolbar | 16 |
| Toolbar Buttons | |
| Shortcut Menus | |
| Login Requirements. | |
| Using Google Search with Smart View | |
| For Microsoft Office 2007 Users. | |

This chapter provides an overview of Oracle Hyperion Smart View for Office, Fusion Edition.

Smart View

Smart View provides a common Microsoft Office interface for Oracle Essbase, Oracle Hyperion Financial Management, Fusion Edition, Oracle Hyperion Planning, Fusion Edition, Oracle Enterprise Performance Management Workspace, Fusion Edition, Oracle Business Intelligence Enterprise Edition, and Oracle's Hyperion® Enterprise® data. Using Smart View, you can view, import, manipulate, distribute and share data in Microsoft Excel, Word, and PowerPoint interfaces.

Smart View Benefits

Smart View benefits:

- A common Microsoft Office interface for Essbase, Financial Management, Planning, Oracle Hyperion Financial Reporting, Fusion Edition, Oracle's Hyperion® Interactive Reporting, Oracle's Hyperion® SQR® Production Reporting, Oracle's Hyperion® Web Analysis, Oracle BI EE, and Hyperion Enterprise
- Ability to import documents from Financial Reporting, Interactive Reporting, Production Reporting, and Web Analysis
- Ability to use smart tags to import Oracle's Hyperion Reporting and Analysis content and functions

Auto deployment to notify users when a newer version of Smart View is available

Smart View Toolbar

The Smart View toolbar in Excel displays buttons (described in Table 1, "Smart View Toolbar Buttons," on page 16) for accessing most Smart View commands. You can customize the toolbar by adding and removing buttons.

- To add or remove buttons from the Smart View toolbar:
- 1 Click on the Smart View toolbar to display a menu of toolbar buttons.
- 2 Click a button to add it to or remove it from the Smart View toolbar. Your selection is displayed on the toolbar immediately.
- 3 Repeat for each button you want to add or remove.

Toolbar Buttons

Table 1 lists the buttons on the toolbar.

Table 1 Smart View Toolbar Buttons

| Button | Purpose | Description |
|----------|---------------------|--|
| | Data Source Manager | Opens the Data Source Manager where you add, delete, and edit data source connections |
| P | Zoom In | Zooms into data from the connected data source |
| P | Zoom Out | Collapses the cell view |
| * | Pivot | Changes the dimension orientation |
| | Keep Only | Deletes all members except the ones that are selected |
| | Remove Only | Removes only the selected members |
| (V) | Refresh | Refreshes data in the active Excel worksheet |
| IK | Submit Data | In Hyperion Enterprise, Financial Management, and Essbase, saves ad hoc data updates to the database |
| Ø | Refresh All | Refreshes data in all worksheets of the Excel workbook |
| Ŋ | Undo | Restores the previous database view |

| Button | Purpose | Description |
|---|--------------------|---|
| CH | Redo | Reverses an Undo |
| | Copy Data Points | Copies data points from Excel that you can paste into Word or PowerPoint |
| | Paste Data Points | Pastes data points that were copied from Excel into Word or PowerPoint |
| .3 | R & A Edit | Enables you to edit Reporting and Analysis documents imported into Excel (not selectable for Hyperion Enterprise) |
| 9 | POV Manager | Opens the POV Manager where you can perform operations on a Point of View |
| | Member Selection | Opens the Member Selection dialog box where you can select members and filter member lists |
| for • | Function Builder | Opens the Function Builder, where you create and validate functions |
| # 1 m 1 m 1 m 1 m 1 m 1 m 1 m 1 m 1 m 1 | Adjust | Provides options for adjusting the values of selected cells |
| 3 | Cell Text | For Hyperion Enterprise and Planning data sources, opens the Cell Text dialog box where you edit supporting text for a range of cells |
| 123 | Supporting Details | For Planning data sources, opens the Supporting Details dialog box where you can provide supplemental calculations for a one-dimensional range of cells |
| ß | Select Form | Opens the Select Forms dialog box where you can select Financial Management, Hyperion Enterprise, or Planning data forms |
| (i) | Instructions | For Planning data sources, displays instructions for data forms |
| H | Take Offline | Opens the Offline Manager (not selectable for Hyperion Enterprise) |
| 2 | Sync Back | Synchronizes data from an offline Planning data form to an online data form (not selectable for Hyperion Enterprise) |
| | Lock | Temporarily locks cell values while Planning enters values in other cells (not selectable for Hyperion Enterprise) |
| + | Expand | Displays all levels of detail for the selected cells (not selectable for Hyperion Enterprise) |
| | Collapse | Collapses all levels of detail for the selected cells (not selectable for Hyperion Enterprise) |
| | Business Rules | Opens the Business Rules dialog box where you select Business Rules for Planning data forms (not selectable for Hyperion Enterprise) |
| * - /× | Rules on Form | Opens the Rules on Form dialog box, where you execute Calculate Form and Calculate Currencies business rules (not selectable for Hyperion Enterprise) |
| 2 | Query Designer | Opens the Query Designer, where you design reports |

| Button | Purpose | Description |
|--------|----------------------------|---|
| | Run Reports | Runs reports designed in the Query Designer |
| B | Visualize in Excel | Enables you to view the detail of a selected Excel data point in a linked report without losing the formatting of the source report |
| R. | Visualize in HVE | Opens Oracle Essbase Visual Explorer (not selectable for Hyperion Enterprise) |
| | Options | Opens the Options dialog box where you set ad hoc, display, and cell style preferences |
| to | Refresh Offline Definition | Enables you to update data on offline data forms with current values from online data forms |
| • | Add or Remove Buttons | Click — opens a menu of Smart View toolbar buttons to add or remove Right-click— opens a menu of toolbars to add or remove |

Shortcut Menus

In Smart View, you can use right mouse clicks to access shortcut menus. If you select a range of cells, use Ctrl+right click to maintain the selection of cells.

Login Requirements

Depending on how Smart View was configured by the adminstrator, you may or may not be required to enter your username and password as you change data providers and Office applications or enter HVE.

Using Google Search with Smart View

With Essbase, you can use the Google search capabilities of Oracle Hyperion Smart Search, Fusion Edition to search your data source for information and display the results on an Excel spreadsheet in Smart View.

Before starting, you must obtain the URL and trigger keyword, which can be provided by the system administrator.

- To use Google with Smart View:
- 1 Open a browser.
- 2 Enter the URL to open. Oracle Hyperion Smart Search, Fusion Edition.
- 3 Enter the trigger keyword.
- 4 In the search box, enter one or more member names separated by spaces.
- 5 Click the link under Search to display the results in Smart View in Excel.

Note:

Excel may display a Security Warning message. If you see this message, you must modify Excel macro security settings to see the search results. To do this, select **Tools**, **then Options** from the Excel toolbar. On the Security tab of the Options dialog box, click Macro Security and set Security Level to medium or low. See the system administrator for more information.

For Microsoft Office 2007 Users

Smart View supports Microsoft Office 2007, whose products feature **ribbons** in place of the previous menu system. In Office 2007 products, the Smart ViewHyperion menu is replaced by the Hyperion ribbon, which contains buttons that access Smart View features. Smart View functionality is the same as it is in previous Office versions; only the method of access is different.

You can still access the Hyperion menu, the online help system, and the Help About screen on the Add-Ins ribbon.

Buttons on the Hyperion ribbon are labeled and grouped according to function.

Table 2 describes Smart View buttons on the Oracle Hyperion Enterprise Performance Management System ribbon, grouped according to ribbon sections.

Table 2 Smart View Buttons on the Hyperion Ribbon for Office 2007

| Button | Description |
|---------------------|--|
| Data Source Manager | Opens the Data Source Manager where you add, delete, and edit data source connections |
| Activate | Enables you to select the data source that you want to activate for the worksheet |
| Reset | Disconnects the worksheet from its current data source |
| Zoom In | Zooms into data from the connected data source |
| Zoom Out | Collapses the cell view |
| Pivot | Changes the dimension orientation |
| Keep Only | Deletes all members except the ones that are selected |
| Remove Only | Removes only the selected members |
| Select Form | Opens the Select Forms dialog box where you can select Financial Management or Planning data forms |
| Instructions | For Planning data sources, displays instructions for data forms |
| Take Offline | Opens the Offline Manager (not selectable for Hyperion Enterprise) |
| Refresh Offline | Enables you to update data on offline data forms with current values from online data forms |
| Sync Back to Server | Synchronizes data from an offline Planning data form to an online data form |
| Lock | Temporarily locks cell values while Planning enters values in other cells (not selectable for Hyperion Enterprise) |

| Button | Description |
|--------------------|--|
| Expand | Displays all levels of detail for the selected cells (not selectable for Hyperion Enterprise) |
| Collapse | Collapses all levels of detail for the selected cells (not selectable for Hyperion Enterprise) |
| Refresh | Refreshes data in the active Excel worksheet |
| Refresh All | Refreshes data in all worksheets of the Excel workbook |
| Submit | In Reporting and Analysis and Essbase, saves ad hoc data updates to the database |
| Undo | Restores the previous database view |
| Redo | Reverses an Undo |
| Copy Data | Copies data points from Excel that you can paste into Word or PowerPoint |
| Paste Data | Pastes data points that were copied from Excel into Word or PowerPoint |
| R&A: Import | Opens the ImportEPM Workspace Document dialog box where you can import Reporting and Analysis documents (not selectable for Hyperion Enterprise) |
| R&A: Edit | Enables you to edit Reporting and Analysis documents imported into Excel (not selectable for Hyperion Enterprise) |
| POV Manager | Opens the POV Manager, where you can perform operations on a Point of View |
| Member Selection | Opens the Member Selection dialog box, where you can select members and filter member lists |
| Function Builder | Opens the Function Builder, where you create and validate functions |
| Adjust | Provides options for adjusting the values of selected cells |
| Cell Text | For Planning data sources, opens the Cell Text dialog box, where you edit supporting test for a range of cells |
| Business Rules | Opens the Business Rules dialog box where you select Business Rules for Planning data forms (not selectable for Hyperion Enterprise) |
| Rules on Form | Opens the Rules on Form dialog box, where you execute Calculate Form and Calculate Currencies business rules (not selectable for Hyperion Enterprise) |
| Supporting Details | For Planning data sources, opens the Supporting Details dialog box, where you can provide supplemental calculations for a one-dimensional range of cells |
| Query Designer | Opens the Query designer, where you design reports |
| Run Report | Runs a report designed in the Query Designer |
| Visualize in Excel | Enables you to view the detail of a selected Excel data point in a linked report without losing the formatting of the source report |
| Visualize in HVE | Opens Visual Explorer (not selectable for Hyperion Enterprise) |
| Options | Opens the Options dialog box where you set ad hoc, display, and cell style preferences |

2

Features and Data Source Providers

In This Chapter

| Ad Hoc Features and Supported Data Source Providers | 21 |
|---|----|
| Query Designer Features and Supported Data Source Providers | 23 |
| Dynamic Data Access Features and Supported Data Source Providers | 24 |
| Hyperion Visual Explorer Features and Supported Data Source Providers | 24 |
| Data Form Features and Supported Data Source Providers | 24 |
| Function Features and Supported Data Source Providers | 25 |
| Reporting and Analysis Document Import Features and Supported Data Source Providers | 26 |

Not all Smart View features pertain to all data source providers. The tables in this chapter list the supported data source providers for each feature.

Ad Hoc Features and Supported Data Source Providers

Ad Hoc functionality is described in Chapter 9, "Working with Ad Hoc Analysis."

| Ad Hoc Feature | Data Source Providers |
|---|---|
| Double-click to invoke the Ad Hoc analysis grid | Essbase, Financial Management, Hyperion Enterprise, Oracle BI EE |
| Display POV member selector | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Drag and drop members from POV to grid | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Set default POV by the user | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Duplicate member name support | Essbase, Financial Management |
| Zoom in and out on dimensions and members | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE(zoom in only) |
| Zoom in and out on Hybrid Analysis members | Essbase |
| Drill-through reports | Essbase |

| Ad Hoc Feature | Data Source Providers |
|---|---|
| Pivot dimensions to columns or rows | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Display cell text | Financial Management, Planning |
| Submit data | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Keep only or remove only certain data cells | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Select attribute dimensions | Essbase, Planning |
| Filter members by UDA | Essbase |
| Filter members by attribute dimensions | Essbase |
| Adjust data values in cells | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Calculate data | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Translate and consolidate data | Financial Management |
| Asymmetric grids and free form mode | Essbase, Financial Management, Planning, Oracle BI EE |
| Save formatting | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Preserve formulas | Essbase, Financial Management, Planning |
| Undo and redo last actions | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Select members | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Select Dynamic Time Series members | Essbase |
| Suppress rows | Essbase, Financial Management, Hyperion Enterprise, Oracle BI EE |
| Indent columns | Essbase, Financial Management, Hyperion Enterprise |
| Navigate without data | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Specify zoom-in levels | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Specify member retention | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Enable or disable double-clicking | Essbase, Financial Management, Hyperion Enterprise |

| Ad Hoc Feature | Data Source Providers |
|---|--|
| Enable or disable undo feature | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Specify labels for missing and no access data cells | Essbase, Financial Management, Hyperion Enterprise, Oracle BI EE |
| Specify member display options | Essbase, Financial Management, Planning |
| Display data status | Financial Management, Hyperion Enterprise |
| Display calculation status and process management level | Financial Management |
| Specify number of decimal places | Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Specify thousands separator | Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Specify ancestor position | Financial Management, Hyperion Enterprise |
| Display messages | Essbase, Financial Management, Hyperion Enterprise |
| Apply cell styles | Essbase, Financial Management, Hyperion Enterprise, Planning |
| VBA function support | Essbase, Financial Management |

Query Designer Features and Supported Data Source Providers

Query Designer functionality is described in Chapter 8, "Query Designer."

| Query Designer Feature | Data Source Providers |
|---|---|
| Select members by using Member Selection | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Drag and drop dimensions from POV toolbar | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Enter members into the grid | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Pivot dimensions | Essbase, Financial Management, Hyperion Enterprise, Planning |
| Filter data | Essbase, Planning |
| Filter members | Essbase, Hyperion Enterprise, Planning |
| Execute MDX queries | Essbase, Planning |
| Modify queries from existing reports | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |

| Query Designer Feature | Data Source Providers |
|------------------------|---|
| Run queries | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |

Dynamic Data Access Features and Supported Data Source Providers

Dynamic data access functionality is described in Chapter 11, "Accessing Dynamic Data Across Microsoft Office."

| Dynamic Data Access Feature | Data Source Providers |
|---|---|
| Copy and paste data points between Excel, Word, and PowerPoint | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Retrieve original Excel grid from data points in Word or PowerPoint | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| View data points in Visual Explorer | Essbase, Planning |
| Change POV in Word or PowerPoint by using POV Manager | Essbase, Financial Management, Hyperion Enterprise, Planning, Oracle BI EE |
| Change URL connection by using POV Manager | Essbase, Financial Management, Hyperion Enterprise, Planning |

Hyperion Visual Explorer Features and Supported Data Source Providers

Visual Explorer functionality is described in Chapter 12, "Visual Explorer."

| Visual Explorer Feature | Data Source Providers |
|---|---|
| View data in graphical format | Essbase, Planning (only with Oracle Hyperion Provider Services) |
| Retrieve Excel grid from graphical data | Essbase, Planning (only with Provider Services) |
| Change graph style | Essbase, Planning (only with Provider Services) |

Data Form Features and Supported Data Source Providers

Data form functionality is described in Chapter 7, "Working with Data Forms" and Chapter 15, "Working with Offline Planning."

| Data Form Feature | Data Source Providers |
|--|---|
| | Financial Management, Hyperion Enterprise, Planning |
| View instructions | Financial Management, Hyperion Enterprise, Planning |
| Adjust data values in cells | Financial Management, Hyperion Enterprise, Planning |
| Submit data | Financial Management, Hyperion Enterprise, Planning |
| Specify decimal places | Financial Management, Hyperion Enterprise, Planning |
| Specify thousands separator | Financial Management, Hyperion Enterprise, Planning |
| Manipulate data by performing unary operations | Financial Management, Planning |
| Lock and spread values to base periods | Planning |
| Enter supporting detail | Planning |
| Enter cell text | Financial Management, Planning |
| Work offline | Planning |
| Synchronize with Planning applications | Planning |
| Execute Business Rules | Planning |
| View Business Rules with runtime prompts | Planning |
| Preserve formulas | Financial Management, Planning |
| Apply cell styles | Financial Management, Hyperion Enterprise, Planning |
| Support Offline Planning | Planning |
| Search page dimensions | Planning |
| Hide header rows | Planning |

Function Features and Supported Data Source Providers

Functions are described in Chapter 14, "Functions."

| Function Feature | Data Source Providers |
|-------------------|--|
| Use HsGetValue | Essbase, Financial Management, Hyperion Enterprise |
| Use HsSetValue | Essbase, Financial Management, Hyperion Enterprise |
| Use HsDescription | Financial Management, Hyperion Enterprise |
| Use HsGetText | Financial Management |
| Use HsSetText | Financial Management |

| Function Feature | Data Source Providers |
|---|--|
| Use HsCurrency | Financial Management, Hyperion Enterprise |
| Use HsLabel | Financial Management, Hyperion Enterprise |
| Refresh active worksheets or workbooks | Essbase, Financial Management, Hyperion Enterprise |
| Migrate Financial Management spreadsheet add-in functions to new syntax | Financial Management, Hyperion Enterprise |
| Display functions by using smart tags | Essbase, Financial Management, Hyperion Enterprise |

Reporting and Analysis Document Import Features and Supported Data Source Providers

Reporting and Analysis document import functionality is described in Chapter 13, "Importing Reporting and Analysis Documents into Microsoft Office."

| Reporting and Analysis Feature | Data Source Providers |
|--|-----------------------|
| Import Reporting and Analysis documents into Microsoft Excel, Word, PowerPoint, and Outlook (if Word is the e-mail editor) | EPM Workspace |
| Edit and refresh Reporting and Analysis documents | EPM Workspace |
| Import multiple-page documents | EPM Workspace |
| Use prompts | EPM Workspace |
| Import Reporting and Analysis documents by using smart tags | EPM Workspace |

3

Installing and Enabling Smart View and Visual Explorer

In This Chapter

| Installing Smart View | 27 |
|---|----|
| Enabling and Disabling Smart View | |
| Smart View and Essbase Spreadsheet Add-in | 28 |
| Information for Administrators | 28 |

Installing Smart View

Because Smart View is an add-in to Microsoft Office products, you can install only on Windows platforms.

To view imported data within Microsoft Office, each user in an organization must install Smart View.

You can install Smart View either through the Oracle Hyperion Enterprise Performance Management System Installer, Fusion Edition or from an individual installation file. For information on installing Smart View from the Oracle Hyperion Enterprise Performance Management System Installer, Fusion Edition, see the EPM System Installation and Configuration Guide.

- To install Smart View from the installation file:
- 1 Navigate to the location of the Smart View installation file specified by your administrator.
- 2 Double-click smartview.exe to start the installation wizard.

If you are licensed to use Visual Explorer, run SmartViewHVE.exe instead.

- 3 Click Next.
- 4 Optional: Click Change to install to a folder other than the default.
- 5 Click Next.
- 6 Click Finish. The next time you open Microsoft Excel, Word, or PowerPoint, the Hyperion menu or ribbon is displayed.

Enabling and Disabling Smart View

- To enable or disable Smart View:
- 1 Select Hyperion, then About.
- 2 Select or clear **Enable Add-in** to enable or disable Smart View.

Your selection takes effect the next time you start a Microsoft Office application.

Smart View and Essbase Spreadsheet Add-in

Smart View and Essbase Spreadsheet Add-in (release 7.1.2 or later only) can be installed on the same computer, but you must enable compatibility between them.

- To enable compatibility between Essbase Spreadsheet Add-in and Smart View:
- 1 Open Excel.
- 2 Select Essbase, then Options, then Global.
- 3 Select Limit to Connected Sheets.
- 4 Click OK.

Note:

You can connect to data sources from Smart View and Essbase in the same workbook but not on the same worksheet.

Information for Administrators

The following information in this section is for system administrators.

- "Running Silent Installations" on page 28
- "Checking for New Smart View Versions" on page 29
- "Compatibility with Data Sources" on page 29
- "Uninstalling Smart View" on page 29

Running Silent Installations

Administrators can enable silent installations — installations that do not require settings to be specified each time — for Smart View. The silent installation command can be included in scripts to automate installation.

- To run silent installations:
- 1 Open a command line prompt.

- 2 Navigate to the directory of the Smart View installer.
- 3 Run one of the following commands:
 - To install in the default directory, run SmartView.exe /s /v"/qn URL=http://localhost:13080/aps/SmartView
 - To install in another directory, run SmartView.exe /s /v"/qn
 INSTALLDIR=<target path>", for example, SmartView.exe /s /v"/qn
 INSTALLDIR=D:\SmartView"

For Visual Explorer users, replace SmartView.exe with SmartViewHVE.exe in the command line.

- To run silent installation and log the installation sequence:
- 1 Open a command line prompt.
- 2 Navigate to the directory of the Smart View installer.
- 3 Run SmartView.exe/s/v"/qn INSTALLDIR=D:\SmartView /L*v c:\install.log".

For Visual Explorer users, replace SmartView.exe with SmartViewHVE.exe in the command line.

Checking for New Smart View Versions

Smart View supports automatic deployment, which detects when a newer version of Smart View is available to install. Automatic deployment upgrades only Smart View releases 9.0 and later.

Compatibility with Data Sources

See the compatibility matrix in the *Smart View Readme* for information on which data source provider versions are compatible with this release of Smart View.

Uninstalling Smart View

Close all Microsoft Office applications, then use Add or Remove Programs on the Windows Control Panel to uninstall Smart View, which is listed as **Oracle Hyperion Smart View for Office**, **Fusion Edition**.

When Smart View is installed, the .MSI file is extracted from SmartView.exe or SmartViewHVE.exe and placed at C:\WINNT\Downloaded Installations\{<a dynamic GUID>}\. The .MSI is cached in this location through InstallShield Integrated Development Environment for maintenance mode activities, such as repairing and removing files.

The uninstallation details are stored at:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall
\<Product GUID>

Under this key, the sub key "InstallSource" holds the value C:\WINNT\Downloaded Installations\{<a dynamic GUID>}\.

The uninstallation process refers to this sub key and locates the extracted .MSI file. Because the .MSI file present in this location only initiates the uninstallation process, you cannot delete the .MSI file during uninstallation. Only after the product is fully uninstalled can you manually delete the .MSI file.

Registry Information

To preserve login and user preference information, the following registry items remain after uninstalling Smart View:

HKEY_CURRENT_USER\Software\Hyperion Solutions\ HyperionSmartView \Connections

HKEY_CURRENT_USER\Software\Hyperion Solutions\HyperionSmartView\Login

HKEY_CURRENT_USER\Software\Hyperion Solutions\ HyperionSmartView \Preferences

Visual Explorer File Association

If Smart View and Essbase Spreadsheet Add-in are installed on the same computer, when you open a Visual Explorer . TWB file directly from Windows Explorer, the file is opened by Smart View, regardless of whether the file was created in Smart View or Spreadsheet Add-in.

If you uninstall Smart View, and keep Spreadsheet Add-in on the computer, the file association for .TWB files is lost, even with Spreadsheet Add-in.

Use Windows Explorer to establish . TWB file association manually with Spreadsheet Add-in.

4

Connecting to Data Sources

In This Chapter

| The Data Source Manager and Connections | |
|--|----|
| Common Provider Connections | |
| Independent Provider Connections | |
| Migrating Essbase Connections to the Data Source Manager | |
| Simulation and Forecasting Workbooks | 42 |
| Changing Passwords | 44 |

The Data Source Manager and Connections

Smart View provides the Microsoft Office interface within which you work with the data that is contained in your data source. How you connect to your Smart View data sources and manage the connections depends on the data source type that you use.

- Common Provider Connections: Essbase, Planning, and Oracle BI EE data sources are connected through a common provider, Provider Services. Connections are managed in the Smart View Data Source Manager. See "Common Provider Connections" on page 32.
- Independent Provider Connections: Financial Management, Reporting and Analysis, and Hyperion Enterprise data sources are connected directly or through Oracle's Hyperion® Shared Services. See "Independent Provider Connections" on page 35.
- Simulation & Forecasting Workbooks: If you use Oracle Crystal Ball, Fusion Edition, you can apply its simulation and forecasting models to data from all supported data sources in an Excel workbook. See "Simulation and Forecasting Workbooks" on page 42

Common Provider Connections, Independent Provider Connection, and Simulation & Forecasting Workbooks are displayed as alternate interfaces in the Data Source Manager, which contains a **Switch** button that enables you to select among them.

Note:

If you are upgrading from a previous version of Smart View, you can migrate your Essbase connections to the Data Source Manager. See "Migrating Essbase Connections to the Data Source Manager" on page 41

A data source consists of the following components, and each data source is unique.

- Data source type (for example Essbase)
- Product server name (Essbase and Oracle BI EE) or server node name (Planning)
- Product server URL (Oracle BI EE and Planning)
- Application
- Database or cube
- Data Form (Planning only)

Common Provider Connections

Data Sources: Essbase, Planning, Oracle BI EE

The Data Source Manager is a multi-functional feature. In addition to managing data source connections, you use it to create and manage Smart Slices and queries (see Chapter 5, "Smart Slices" and Chapter 6, "The Report Designer"), open data forms or initiate ad hoc analysis (see Chapter 7, "Working with Data Forms" and Chapter 9, "Working with Ad Hoc Analysis"). This section describes Data Source Manager common provider connection functions:

- "Connecting to Common Provider Data Sources" on page 32
- "Disconnecting Common Provider Connections" on page 33
- "Setting Data Source Manager Views" on page 33
- "Associating a Connection to a Worksheet" on page 33
- "Adding and Deleting Connections" on page 34
- "Setting Provider Properties" on page 34

Connecting to Common Provider Data Sources

- To connect to a data source:
- 1 Select Hyperion, then Data Source Manager.
- 2 From Smart View Data Source Manager, click Connect to Provider Services.

The Data Source Manager displays an expandible tree list of available data sources nested as follows:

```
Data Source type
Server
Application
Database or Cube
Smart Slice
```

- 3 Expand the data source type and select a server.
- 4 In **Connect to Data Source**, enter your user name and password to the selected database. Contact the system administrator if you do not have this information.
- 5 Click Connect.

You are connected to data sources down to the Smart Slice level. With these data sources, you can perform any Smart View operation valid for a given data source.

Disconnecting Common Provider Connections

In the Data Source Manager, you can disconnect from the data source or from Provider Services, depending on the object selected.

- To disconnect in the Data Source Manager:
- 1 Right-click the object to disconnect in the Data Source Manager tree.
- 2 Select **Disconnect**, then one of the following:
 - Data Source Connection to disconnect from the data source.
 - Common Provider Connection to disconnect from Provider Services, the data provider for the data source.

Setting Data Source Manager Views

To facilitate finding common provider data sources in the Data Source Manager tree list, you can specify how data sources are displayed from the Views button:

- All displays all data sources for each server in the tree view, based on user permission..
- **Pre-defined** displays all data sources that have been specified by the Administrator for inclusion in the Data Source Manager.
- Favorites displays data sources added to the list by users from the Pre-defined or All views...
- To select data sources to be displayed when Pre-defined or Favorites is selected, the data sources to be displayed in the Data Source Manager:
- 1 From the Data Source Manager toolbar, select **Views** then **All**.
- 2 From the tree list, select a data source.
- 3 Select Add to View then Pre-defined or Favorites.
- 4 Repeat as necessary.
- 5 Click Sync with Smart Slice.

Associating a Connection to a Worksheet

To associate a connection with a particular worksheet, the connection must be included in the Favorites view (see "Setting Data Source Manager Views" on page 33).

- To associate a connection to a worksheet:
- 1 Expand the Data Source Manager tree and select a database that is in the Favorites view.

2 Select Hyperion, then Activate. The connection is displayed in the Active Connections list available from the Hyperion menu.

Adding and Deleting Connections

You must have administrative privileges for a data source —Essbase, Planning, or Oracle BI EE — to add or delete connections in the Data Source Manager.

- To add a connection in the Data Source Manager:
- 1 Select Hyperion then Data Source Manager.
- 2 From Smart View, click Connect to Provider Services.
- 3 From the Data Source Manager tree view, select a data source type, for example, Essbase.
- 4 In Check User Authorization, enter the following information:
 - Product—the data source type (Essbase, Planning, or Oracle BI EE)
 - **Product Server Name**—the name of the computer on which the data source server is located (Essbase) or data source connection name (Oracle BI EE or Planning)
 - Server Node Name—the data source type (Planning only)
 - Product Server URL—the URL of the server, using the syntax jdbc:oraclebi://
 <servername>:9703/ for Oracle BI EE or http://<servername>:8300/
 HyperionPlanning/SmartView for Planning for Planning
 - User Name—the name of the user with Administrator privileges to add data sources.
 - Password—the password of the Administrator to the server.
- 5 Click OK.

You can specify how data sources are displayed as described in "Setting Data Source Manager Views" on page 33.

- To delete a data source connection:
- 1 From the Data Source Manager, select a server name.
- 2 From the toolbar, click the **Delete** button.

Setting Provider Properties

Administrators can set properties for Provider Services to apply to common provider connections in the Data Source Manager. Select a database, or data form in the Data Source Manager, then click **Provider Services Properties** and specify properties as described in Table 3.

Table 3 Provider Properties

| Property | Description |
|-----------------------|---|
| sessionTimeOut | the period of inactivity (in minutes) before a user session times out. |
| numberOfRows | the maximum number of rows permitted in a Smart View grid. |
| numberOfColumns | the maximum number of columns permitted in a Smart View grid. |
| forceClientUpgrade | enables forced upgrade of Smart View when a newer version is available |
| warnClientUpgrade | enables a message to upgrade to a newer version of Smart View when the newer version is available |
| enableProviderLogging | enables Provider Services to log errors and information. |
| logFile | if enableProviderLogging is set to True, the location where the log is stored. |
| logLevel | if enableProviderLogging is set to True, the level of errors that are logged |
| enableProviderTracing | enables log tracing |

Independent Provider Connections

Data Sources: Financial Management, Reporting and Analysis, Hyperion Enterprise

- "Adding a Data Source Through Direct Connection" on page 36
- "Adding a Data Source Through Shared Services" on page 38
- "Associating a Data Source Connection to a Worksheet" on page 39
- "Editing a Data Source Connection" on page 39
- "Setting a Default Connection" on page 40
- "Deleting a Data Source Connection" on page 41
- "Disconnecting from a Data Source" on page 41
- "Checking for Active Connections" on page 41
- "Disconnecting a Worksheet from its Data Source" on page 41

About Independent Provider Connections

Data Sources: Financial Management, Reporting and Analysis, Hyperion Enterprise, Planning

Independent provider connections are not specific to worksheets and you can establish multiple connections per Excel instance. A connection is a communication line between the user and data source regardless of whether the data source is connected to or disconnected from the provider server.

Note:

Planning data source connections can be made through eitherProvider Services or the Planning provider. However, using the Planning provider does not give the benefits of ad hoc analysis, Smart Slices, or the Report Designer. This option is provided for organizations who would like to upgrade to Release 11.1.1, but would like to plan the move to the common provider.

Adding a Data Source Connection

The data sources to which you can connect are displayed in the Data Source Manager. Financial Management, Hyperion Enterprise, Planning, Financial Reporting, Interactive Reporting, Production Reporting, and Web Analysis. You can connect to a data source directly using a URL or through Shared Services.

Note:

Hyperion Enterprise does not support Shared Services.

You need to know the following information to add a data source connection:

- If you are working with a database or repository, the location of the URL where the data source is hosted
- If you are working offline, the path of the local storage directory where your metadata resides
- The user name and password that provides you access to the data source, whether it is a URL or directory location

If you do not know this information, contact your system administrator.

For more information on adding a data source, see:

- "Adding a Data Source Through Direct Connection" on page 36
- "Adding a Data Source Through Shared Services" on page 38

Adding a Data Source Through Direct Connection

To add a data source connection directly, you need to know the URL for the data source provider. You can get this information from your Administrator.

- To add a data source connection through direct connection:
- Select Hyperion, then Data Source Manager. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.
- 2 In Independent Provider Connections, click the Add button.
- 3 Select URL Provider.

The Add Connection - Provider Type/URL dialog box is displayed.

4 From **Provider**, select the type of data source to which you want to connect:

- Hyperion Provider (for Financial Management, Hyperion Enterprise, and Planning)
- Hyperion Reporting and Analysis System 9 Provider (for Financial Reporting, Interactive Reporting, Production Reporting, and Web Analysis)
- In the Location drop-down list box, type the URL or the local storage directory for the data source to which you want to connect, and then click Next.

The URL syntax for the various data sources is as follows. Contact your system administrator for the URL to use:

```
Financial Management: http(s)://<servername>/hfmofficeprovider/hfmofficeprovider.aspx
```

```
Hyperion Enterprise: http://<servername>/heofficeprovider/
heofficeprovider.aspx
```

Planning: http(s)://<servername>:8300/HyperionPlanning/SmartView

Reporting and Analysis: http(s)://<servername>:19000/workspace/browse/listxml

If you are accessing a local storage directory, you can click Browse to open the Browse for Folder dialog box. Navigate to the location where your metadata resides, and click OK.

Note:

You can click the Back button at any time to return to the previous dialog box and edit the entries you already made.

6 In the Add Connection - Application/Cube dialog box, expand the Servers node.

A list of servers that are accessible from the URL you selected in step 5 is displayed.

7 Expand the nodes until the object to which you want to connect is displayed, then select that object.

Note:

You may be prompted to enter your login information if you try to expand an application for which you do not have access.

- 8 In the Connect to Data Source dialog box, type the user name and password, and then click Connect.
- 9 Select the application and click Next. In the Add Connection Name/Description dialog box, type a name for this data source connection in the Name text box.

The data source connection name is also referred to as the friendly name.

Note:

Do not use semicolons (;) in connection names planned for using functions. An error occurs when using Function Builder to create functions and when pasting data points containing functions.

10 In the **Description** text box, type any information that is helpful to you in identifying this data source, and then click **Finish**.

The data source connection you just added is now listed in the Data Source Manager. To connect to the data source, follow the steps in "Connecting to a Data Source" on page 39. To associate the data source to the current worksheet, follow the steps in "Associating a Data Source Connection to a Worksheet" on page 39.

Adding a Data Source Through Shared Services

Note:

This release of Hyperion Enterprise does not support Shared Services.

You can connect to any provider that has been registered with Shared Services as a provider for Smart View content. Adding data sources through Shared Services conveniently provides users with a single location to choose a list of available Smart View content providers.

- ➤ To add a data source through Shared Services:
- Select Hyperion, then Data Source Manager. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.
- 2 In Independent Provider Connections , click the Add button.
- 3 Select Shared Services Provider.

The Add Connection From Shared Services dialog box is displayed.

4 In the **Hyperion Shared Services URL** text box, type the URL to the Shared Services server. The URL syntax for Shared Services is as follows:

```
http://<sharedservices_server>:58080
```

Contact your system administrator for the Shared Services server name.

5 Click Update.

A list of available data sources that are registered with Oracle's Hyperion® Shared Services is displayed in the Select a Connection from a Provider list box.

- 6 From the Select a Connection from a Provider list box, select the data source to which you want to connect.
- 7 Click Next.
- 8 In the Connection credentials dialog box, type the user name and password that you will use to access this data source, and then click Connect.
- 9 In the Add a Connection Name dialog box, type a name for this data source connection in the Name text box.

The data source connection name is also referred to as the friendly name.

10 In the **Description** text box, type any information that is helpful to you in identifying this data source, and then click **Finish**.

The data source connection you just added is now listed in the Data Source Manager. To connect to the data source, follow the steps in "Connecting to a Data Source" on page 39. To associate

the data source to the current worksheet, follow the steps in "Associating a Data Source Connection to a Worksheet" on page 39.

Connecting to a Data Source

Connecting to a data source enables you to retrieve data into Excel.

- To connect to a data source:
- Select Hyperion, then Data Source Manager. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.
- In Independent Provider Connections, select the data source to which you want to connect and click Connect.
- 3 In the Connect to Data Source dialog box, enter the user name and password for the data source.
- 4 Click Connect.
- 5 Click Close.

To associate the data source to the current worksheet, follow the steps in "Associating a Data Source Connection to a Worksheet" on page 39.

Associating a Data Source Connection to a Worksheet

After adding data sources to the Data Source Manager, you need to associate a worksheet with a data source before you can access the data. You can connect to one data source per worksheet.

- To associate a data source to a worksheet:
- 1 In Independent Provider Connections, select a data source and double-click.
- 2 In the Connect to data source dialog box, enter the password and click Connect.

The worksheet is now connected to the selected data source.

Editing a Data Source Connection

When you edit a data source, you can change the provider, change to a different server with the same data source provider, or change the application or repository associated with the data source. You must disconnect from a data source before you can edit it.

- To edit a data source:
- 1 Select **Hyperion**, then **Data Source Manager.** For Office 2007, click **Connect** in the **Connections** section of the **Hyperion** ribbon.
- 2 In the Data Source Manager dialog box, select the data source that you want to modify and click Edit.

The Edit Connection - Provider Type/URL dialog box is displayed.

- 3 In the Provider drop-down list box, select the type of data source to which you want to connect:
 - Hyperion Provider (for Essbase, Financial Management, Hyperion Enterprise, and Planning)
 - Hyperion Reporting and Analysis System 9 Provider (for Financial Reporting, Interactive Reporting, Production Reporting, and Web Analysis)
- 4 In the Location field, edit the URL of your data source or select from the list of available data sources.

The URL syntax for the various data sources is as follows. Contact the system administrator for the URL to use:

Essbase: http(s)://<servername>:13080/aps/SmartView

Financial Management: http(s)://<servername>/hfmofficeprovider/hfmofficeprovider.aspx

Hyperion Enterprise: http://<servername>/heofficeprovider/
heofficeprovider.aspx

Planning: http(s)://<servername>:8300/HyperionPlanning/SmartView

Reporting and Analysis: http(s)://<servername>:19000/workspace/browse/listxml

- 5 Click Next.
- 6 In the Connect to Data Source dialog box, enter the user name and password for the data source, and click Connect.
- 7 In the Edit Connection Application/Cube dialog box, expand the Servers node, select an application, and click Next.

If you do not see any applications, contact your system administrator. You may be prompted to enter your login information if you try to expand an application for which you do not have access.

- 8 In the **Edit Connection Name/Description** dialog box, edit the name and description for your connection user name.
- 9 Click Finish.
- 10 Click Close.

Setting a Default Connection

The default data source is the data source that is automatically used to retrieve data into the Excel worksheet. The default data source will always be the data source used when navigating from worksheet to worksheet. In the Data Source Manager, the data source name listed in bold text is the default connection.

- To set the default data source:
- 1 In Independent Provider Connections, select an existing data source connection.
- 2 Right-click and select Set As Default.

Deleting a Data Source Connection

You must disconnect from a data source before deleting it.

- To delete a data source:
- Select Hyperion, then Data Source Manager. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.
- In the Independent Provider Connections, select the data source that you want to remove and click Delete.
- 3 When prompted to confirm deletion of the data source, click Yes.
- 4 Click Close.

Disconnecting from a Data Source

Disconnecting from a data source means disconnecting from a live connection to the server. The data source connection is still displayed in the Data Source Manager until you delete it. Disconnect from a data source when you do not need to retrieve data from it.

- To disconnect from a data source:
- Select Hyperion, then Data Source Manager. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.
- 2 In Independent Provider Connections, select the data source you want to disconnect from and click Disconnect.
- 3 Click Close.

Checking for Active Connections

To check for all active connections in your Excel session, select **Hyperion**, then **Active Connections**. The data sources to which you are connected are listed.

Disconnecting a Worksheet from its Data Source

To disconnect a worksheet from its current connection to a data source, select **Hyperion**, then **Reset Connection**.

Migrating Essbase Connections to the Data Source Manager

Data Sources: Essbase

With release 11.1.1, Essbase connections from previous releases can be migrated to the Data Source Manager. Connections are migrated to the Favorites mode of the Data Source Manager and retain all connection properties, including connection friendly names.

Migration proceeds whether or not the connection is valid. If the specified server does not exist, for example, or if the application or database are invalid, the connection with all its properties migrates nevertheless.

Note:

In earlier releases of Smart View, it was possible to have multiple connections to the same server-application-database, each with a different name. In the Data Source Manager, however, data sources are unique and cannot have more than one name. If you try to migrate multiple connections to the same data source, only the first one will succeed, and only if the Data Source Manager does not already contain a connection to that data source.

- To migrate an Essbase connection to the Data Source Manager:
- 1 Ensure that Smart View is connected to Provider Services.
- 2 In Independent Provider Connections, right-click the connection to be migrated.
- 3 In Independent Provider Connections, select Migrate to Data Source Mgr.

Simulation and Forecasting Workbooks

Data Sources: Essbase, Oracle BI EE, Planning, Reporting and Analysis, Hyperion Enterprise, Financial Management

Note:

To use Simulation and Forecasting Workbooks, you must be licensed for Crystal Ball, Fusion Edition, which is the only version of Crystal Ball Smart View.

Note:

The operations that you can perform from the Simulation & Forecasting Workbooks panel of the Data Source Manager are determined by the permissions set for you by the EPM Workspace administrator on files in the folder designated as the model repository in EPM Workspace.

Simulation and forecasting workbooks are stored in a centralized EPM Workspace repository and can be managed through the Data Source Manager.

In the Data Source Manager, you access the Simulation and Forecasting Workbook repository, from which you can open Excel workbooks to work with the models. These workbooks may contain one or more worksheets with a Crystal Ball model and one or more other worksheets, each of which may draw data from any of the data sources supported by Smart View.

For more information on Crystal Ball, see the Oracle Crystal Ball User Manual.

Working With Simulation and Forecasting Workbooks

- To work with data in a Crystal Ball workbook:
- 1 Log in to Smart View and select **Hyperion**, then **Data Source Manager**.
- 2 Click Switch and select Simulation & Forecasting Workbooks.
- 3 Click Connect to Repository and if requested, log in to Crystal Ball. A tree list containing the workbooks for which you have permission is displayed.
- 4 Double-click the workbook you want to open. Crystal Ball is automatically loaded.
- 5 Perform Crystal Ball operations as described in Oracle Crystal Ball User Manual.
 - Oracle recommends keeping the Crystal Ball model on a worksheet separate from data source worksheets.
- 6 Click **Submit Data** if you want to submit the new data to the database.

Managing Simulation and Forecasting Workbooks

Use the Data Source Manager to manage Simulation and Forecasting Workbooks. To perform the operations described in this section, you must have permission to do so in EPM Workspace.

Adding and Deleting Workbooks

- To add a workbook to the repository:
- 1 Click
- In the file selection box, navigate to the Excel file to add to the Oracle Crystal Ball, Fusion Edition repository and click OK.
- 3 Click Refresh to display the new workbook in the tree list.

To delete a workbook from the repository, select the workbook from the tree list and click **Delete**.

Setting Properties

The properties that you set apply across all sessions running on the server.

- To specify where workbook files should be stored and which EPM Workspace agent with which to communicate:
- 1 In Simulation & Forecasting Workbooks, select a workbook from the tree list.
- 2 Click Set Options.
- 3 In **Repository Options**, enter the following information:
 - URL: the Web Services agent URL

- Server: the global service manager host name
- Port: the global service manager port number
- Folder: the name of the repository folder to contain the workbook file
- 4 Click OK.

Saving Workbooks from EPM Workspace

- To retrieve a workbook from EPM Workspace and save it to your computer:
- 1 In Simulation & Forecasting Workbooks, select a workbook from the tree list.
- 2 Click and navigate to a local directory to save the workbook.
- 3 Click OK.

Changing Passwords

Data Sources: Essbase, Financial Management

You can change your data source passwords from Common Provider Connections for Essbase and from Independent Provider Connections for Financial Management.

Changing Essbase Passwords

To change a password for Essbase, from Common Provider Connections, right-click the server whose password you want to change and select Change Provider Password. In Change Password for, enter your current password, then enter and confirm your new one. Click Change Password.

Note:

You can use Smart View to change your password only for standalone Essbase computers, not for clusters. Passwords can be changed for clusters in Oracle Essbase Administration Services Console.

Changing Financial Management Passwords

To change a password for Financial Management, fromIndependent Provider Connections, select the server whose password you want to change. From the toolbar, click the Change Password button. In Change Password, enter your current password, then enter and confirm your new one. Click Change Password.

Smart Slices

5

In This Chapter

| Creating Queries from Smart Slices | 45 |
|---|-----|
| Smart Slices, Ad Hoc Analysis, and Data Forms | 46 |
| Creating Smart Slices | 4.7 |

A Smart Slice is a reusable perspective of an Essbase, Planning, or Oracle BI EE data source. It is composed of a single member, a combination of single members, filters, or combination of single members and filters in any order. These components serve as boundaries to the data that users can view and work with in the Smart Slice. In addition, a set of preferences are specified for the Smart Slice. Any operation that can be done in Smart View can be done within the confines of a Smart Slice. Smart Slice boundaries can be as inclusive or exclusive as required to make the Smart Slice useful for data analysis.

An organization can have as many different Smart Slices as it needs to accommodate the specific data requirements of its users. For example, Smart Slices can be created for different sales geographical regions, different product lines, different time frames, or a combination of any of these dimensions.

Smart Slices are created by administrators and stored in the Provider Services, where they are available to all users with the proper privileges.

- "Creating Queries from Smart Slices" on page 45 (all Essbase, Planning, Oracle BI EE users)
- "Creating Smart Slices" on page 47 (administrators only)

Creating Queries from Smart Slices

Data sources: Essbase, Planning, Oracle BI EE

You can do anything in a Smart Slice that you can do in Smart View—perform ad hoc analysis, use free form, submit data, and so forth. You can view and work with any data within the boundaries of a Smart Slice, but not with data outside its boundaries. For example, in a Smart Slice that limits sales data to the Western region, you could drill down to data for California or Los Angeles, but you cannot navigate across to New York.

Smart Slices are particularly useful as the basis for creating queries in the Report Designer (see Chapter 6, "The Report Designer"), where you can design reports based on an entire Smart Slice or on a subset of data in the Smart Slice—a subquery.

Creating Queries from Smart Slices

- To create a query from a Smart Slice:
- 1 From the Data Source Manager, select a Smart Slice.
- 2 Click Insert Into Report and select Query.

The query is displayed in the Report Designer.

3 Design a report for the query as described in Chapter 6, "The Report Designer".

Creating Subqueries from Smart Slices

- To create a subquery:
- 1 From the Data Source Manager, select a Smart Slice.
- 2 Click Insert Into Report and select Subquery.

The Smart Slice designer is displayed.

- 3 From the Smart Slice designer, select members for the data that you want to include in your subquery.
- 4 In Enter a new name, enter a name for the subquery, then press Enter.

The subquery is displayed in the Report Designer.

5 Design a report for the query as described in Chapter 6, "The Report Designer".

Smart Slices, Ad Hoc Analysis, and Data Forms

Smart Slices and Ad Hoc Analysis

To use a Smart Slice with ad hoc analysis (Essbase, Planning, Oracle BI EE) in Excel, select the

Smart Slice in the Data Source Manager, click and select Ad Hoc Analysis. Data and POV from the Smart Slice is entered into the worksheet, and you can perform ad hoc analysis as described in Chapter 9, "Working with Ad Hoc Analysis".

If you want to locate the Smart Slice source of the data in an ad hoc grid, refresh the data and click. The Smart Slice is highlighted in the Data Source Manager.

Planning Data Forms

To use a Planning data form, select the Smart Slice in the Data Source Manager, click and select **Open Form**. See Chapter 7, "Working with Data Forms".

Note:

Only forms enabled by the Planning administrator may be used for ad hoc analysis.

Creating Smart Slices

Data sources: Essbase, Planning, Oracle BI EE

Note:

You must have Administrator privileges for the relevant data source to create a Smart Slice.

Creating Smart Slices involves Setting Smart Slice Data Boundaries and Setting Smart Slice Preferences.

Setting Smart Slice Data Boundaries

- To create a Smart Slice:
- 1 From the Data Source Manager, select the data source on which to base the Smart Slice and click



- 2 In the **Admin Console**, enter the following information:
 - **Product:** the data source type (Essbase, Planning, Oracle BI EE)
 - Product Server Name (Essbase)
 - Product Server URL (Planning, Oracle BI EE)
 - Username
 - Password
- 3 Select an application and click

The Smart Slice designer is displayed.

- 4 From Select Alias Table, select an alias table from the list and click OK.
- From the Smart Slice Designer, use any or all of the following actions to create boundaries for the Smart Slice.
 - Select columns. Click an item under Columns to open Member Selection where you select
 only the members that you want as columns. Click the Filters button and select Children or
 Descendents filters for any of the selected members. Users can drill down on the member
 in the grid only if one of these filters is selected.
 - Select rows. Click an item under Rows to open Member Selection where you select only the members you want as rows. Click the Filters button and select Children or Descendents filters for any of the selected members. Users can drill down on the member in the grid only if one of these filters is selected.
 - Move POV items to Rows or Columns.
 - Restrict POV list items. Select a member in the dropdown menu of a POV item to open Member Selection where you select specific members.

- Remove POV items. Drag the item to be removed from the POV list to the **Fields** area (Planning and Oracle BI EE only).
- Set Options as described in "Setting Smart Slice Preferences" on page 48.
- 6 Click Done. Member Selection is displayed.
- 7 In Member Selection, select a dimension member to use as the default POV and click OK.
- 8 In the Data Source Manager, in **Enter a new name**, enter a name for the Smart Slice.
- 9 Click OK. The Smart Slice is displayed in the tree view of the Data Source Manager under its data source.

Setting Smart Slice Preferences

The preferences that you specify are stored as part of the Smart Slice definition, and they override the global preferences set in the Options dialog box. Users can select options in the Options dialog box that are not grayed out. For more complete information on options, see Chapter 16, "User Preferences" From the Smart Slice Designer, select Options, then select from the dropdown menu or enter preferences:

Table 4 Smart Slice Preferences

| Name | Value | Data Source |
|------------------------------|---|-------------------|
| row_suppression: missing | Disabled: displays data rows containing cells with missing data | Essbase, Planning |
| | Enabled: suppresses data rows that contain cells with missing data | |
| row_suppression: emptyBlocks | Disabled: displays data rows containing cells with missing data | Planning |
| | Enabled: suppresses data rows that contain cells with missing data | |
| col_suppression: missing | Disabled: displays data columns containing cells with missing data | Planning |
| | Enabled: suppresses data columns that contain cells with missing data | |
| zoomin: ancestor | bottom: retrieves data for the lowest members in a dimension | Essbase, Planning |
| | top: retrieves data for the highest members in a dimension | |
| zoomin: mode | children: retrieves data for the children of the current member | Essbase, Planning |
| | descendents: retrieves data for the descendents of the current member | |
| | base: retrieves data for level 0 members | |
| includeSelection: val | Enabled: retains the selected member along with the other members retrieved as a result of the zoom | Essbase, Planning |

| Name | Value | Data Source |
|--------------------------------|---|---------------------------------|
| repeatMemberLabels: supression | Enabled: suppresses repeated member names | Essbase, Oracle BI EE |
| | Disabled: displays repeated member names | |
| missingLabelText: val | The text to be displayed in cells that contain missing data | Essbase, Planning, Oracle BI EE |
| noAccessText: val | The text to be displayed in cells containing data not permitted to the user | Essbase |
| aliasTableName: val | Displays the selected alias table of the Smart Slice | Essbase, Planning |
| essIndent: val | None | Essbase, Planning |
| | Subitems: indents only descendents | |
| | Totals: indents only ancestors | |
| numFormat: val | A sample number that displays your selections for numFormat: thousands separator and numFormat: decimal places. | Essbase, Planning, Oracle BI EE |
| numFormat: thousands separator | Yes: use a thousands separator | Essbase, Planning, Oracle BI EE |
| | No: do not use a thousands separator | |
| numFormat: decimal places | The number of decimal places data displays | Essbase, Planning, Oracle BI EE |

6

The Report Designer

In This Chapter

| Creating Reports | 51 |
|------------------------------------|----|
| Controlling Reports. | 53 |
| Deleting Reports | 54 |
| Cascading Reports and Ad Hoc Grids | |

The Report Designer is a Smart View feature in which you can author and manage a variety of reports that are based on Smart Slices. The Report Designer can be used with Essbase, Planning, or Oracle BI EE data sources. For Financial Management, Reporting and Analysis, and Hyperion Enterprise, use the Query Designer as described in Chapter 8, "Query Designer" or functions as described in Chapter 14, "Functions".

Creating Reports

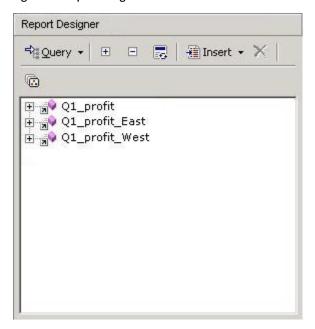
Data sources: Essbase, Planning, Oracle BI EE

You design reports based on Smart Slices from the Report Designer interface, which is displayed when you insert a Smart Slice query or subquery from the Data Source Manager. Reports can then be displayed on an Excel spreadsheet, Word document, or PowerPoint slide. You can display as many reports from as many data sources as space will permit on one sheet. A report can be displayed as a function grid, a table, or a chart (see "Report Types" on page 52).

An entire report is associated with an Excel workbook, a Word document, or a PowerPoint presentation. One report is associated with an Excel worksheet, a Word page, or a PowerPoint slide. For PowerPoint presentations, Oracle recommends one report type per slide.

Figure 1 shows a Report Designer containing three queries inserted from the Data Source Manager. In the example shown, Q1_profit is a query based on an entire Smart Slice and the other two are subqueries created from Q1_profit.

Figure 1 Report Designer



Report Types

This section describes the types of report you can create — function grids, tables, and charts.

Function Grid

Function grids display query results in a dynamic grid format. Function grids can be used with Word, PowerPoint, and Excel. Characteristics of each cell in the grid are displayed when you click or mouse over the cell. To place a function grid on the worksheet, click Insert, select Function Grid, then Refresh.

When you refresh a function grid, data cells are refreshed; members are not. To refresh both data and members, you must regenerate the function grid by reinserting in into the sheet. For this reason, function grids are most useful for reports in which members remain reasonably static. For reports whose members may change more often, tables and charts are better report types.

Although you can have multiple reports on a worksheet, you can have only one function grid.

Formula Retention in Function Grids

You can use Excel formulas, for example SUM, with function grids. To retain such formulas as part of the function grid, you must leave one empty row between the grid and the cell containing the formula and include the empty row in the range of cells selected for the formula definition. This permits retention of the formula when refreshing the data results in a different number of rows in the grid.

Formatting Function Grids

To format a function grid, use Excel formatting capabilities.

Table

Table reports display results in a grid format that floats on the document and can be moved and resized. (To move tables, you must be in Excel design mode, available from the Excel Developer ribbon or Visual Basic toolbar.) Tables are useful for displaying large grids in a smaller space; their scrollbars enable you quickly to access rows and columns. Tables can be used with PowerPoint and Excel.

You can zoom in and out in a table report, but you cannot perform other ad hoc operations or use free form.

To place a scrollable table on the worksheet, click Insert and select Table, then Refresh.

When you refresh a table, both members and data are refreshed.

Chart

Chart reports display results in a chart format that floats on the document and can be moved and resized. (To move charts, you must be in Excel design mode, available from the Excel Developer ribbon or Visual Basic toolbar.) Charts can be used with PowerPoint and Excel. To place a chart on the worksheet, click **Insert** and select **Chart**, then **Refresh**.

When you refresh a chart, both members and data are refreshed.

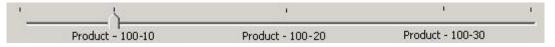
Note:

In PowerPoint, contents of charts and tables are visible only in presentation mode.

Controlling Reports

Queries inserted in the Report Designer cannot be modified. However, you can control the POV of a query in any of its reports that are displayed on the sheet using the slider shown in Figure 2. The slider displays a selected set of dimension members from a query; when you drag the slider marker to a member, its data is displayed in all reports associated with the query on the sheet. Sliders can contain dimensions from more than one query in the Report Designer if the dimensions have the same boundaries.

Figure 2 Slider



Creating a Slider from One Query

- To create a slider:
- Ensure that one or more report type is inserted in the worksheet for the query for which you want to create the slider.
- 2 Click Query View and select Query View
- 3 In the Report Designer, select the query on which to base the slider.
- 4 Click Insert and select Slider to open Member Selection.
- 5 Select a dimension, members, and filters for the slider and click OK.

The slider is displayed on the sheet.

Note:

To move or resize a slider, you must be in Excel design mode, available from the Excel Developer ribbon or Visual Basic toolbar

Creating a Slider from Joined Queries

You can create a slider that contains dimensions from multiple queries if, and only if, the dimensions from the different queries have the exact same boundaries.

- To create a slider using a dimension from multiple queries:
- Ensure that one or more report type is inserted in the worksheet for the query for which you want to create the slider.
- 2 Click Query View and select Dimension View. Notice that the Report Designer tree view is grouped by dimensions rather than by queries. Under each dimension are the queries that contain that dimension. If the dimensions do not contain the same boundaries, multiple sliders will be created to accomodate each of them. For example, if the Market dimension in one query contains a children filter and the Market dimension from another query contains a descendents filter, two Market sliders would be created.
- 3 In the Report Designer, select the dimension on which to base the query.
- 4 Click and select Slider to open Member Selection.
- 5 Select dimension members, and filters for the slider and click OK.

The slider is displayed on the sheet.

Deleting Reports

To delete objects, you must be in Excel design mode, available from the Excel Developer ribbon or Visual Basic toolbar. To delete a whole function grid, table, or chart from a document, select the object and press the Delete key. The deletion is reflected in the Report Designer.

Cascading Reports and Ad Hoc Grids

You can create separate reports for any or all of the members of one dimension in a report based on a Smart Slice query in the Report Designer or an ad hoc grid and cascade these reports separately across the worksheets of an Excel workbook. For reports created in the Report Designer, you can also cascade reports across slides in a PowerPoint presentation (worksheets or slides are created as needed to accommodate all reports).

Cascading Smart Slice Reports

Formulas, comments and other text, function grids, charts, tables, and sliders are included in cascaded reports.

- To cascade a report based on a query in the Report Designer:
- 1 Select a query or subquery in the Report Designer.
- 2 From the Report Designer, click to open Member Selection.
- 3 Select the POV dimension to use as the basis for the report.
- 4 Select all members of the dimension selected in step 3 for which you want to create reports. One report will be generated for each member you select.
- 5 Click **OK** to begin cascading.

A new Excel workbook or PowerPoint presentation is created for the cascaded report, leaving the original intact. In Excel, each worksheet tab is named for the dimension and member of the report it contains.

Cascading Ad Hoc Grids

Formulas, comments and other text, grids, and charts are included in cascaded reports.

- To cascade an ad hoc report:
- Open or create an ad hoc grid on an Excel worksheet (see Chapter 9, "Working with Ad Hoc Analysis" for more information).
- 2 From the Data Source Manager, click it to open Member Selection.
- 3 Select the POV dimension to use as the basis for the report.
- 4 Select all members of the dimension selected in step 3 for which you want to create reports. One report will be generated for each member you select.
- 5 Click **OK** to begin cascading.

A new Excel workbook is created for the cascaded report, leaving the original intact. Each worksheet tab is named for the dimension and member of the report it contains.

Note:

For both types of cascading, to enable worksheet tab naming, do not use more than 31 characters or any the following characters for dimension or member names: (): $\/?*[$].

7

Working with Data Forms

In This Chapter

| Working with Data Forms in Excel Worksheets | 57 |
|---|----|
| Submitting Data | 68 |
| Working with Business Rules for Planning | 69 |
| Adjusting and Spreading Data Values | 72 |
| Working With Supporting Detail | 79 |
| Working with Formulas in Data Forms | 83 |
| Working with Smart Lists | 84 |

This chapter explains basic Smart View concepts for working with data forms. Smart View enables you to export Planning, Financial Management, and Hyperion Enterprise data forms to Microsoft Excel. Using Smart View, you can work with data forms either online connected to the Planning or Financial Management or Hyperion Enterprise server, or offline disconnected from the server.

Working with Data Forms in Excel Worksheets

You can use Excel functionality with Planning, Financial Management, and Hyperion Enterprise data forms. You can view data form data and instructions, create formulas, and format the color and text in a grid. You cannot create formulas in Hyperion Enterprise data forms. Financial Management supports data forms, but the Financial Management Add Member functionality is not supported in data forms opened in Smart View.

When you import a data form into Excel, Smart View retains the customization for the data form. Although you cannot modify the data form structure, you can modify data values in the data form. You can create numeric analyses and calculations for the data, which are preserved in the worksheet when the data form is brought back to the server. This feature provides faster analytic capability, especially when you work with data forms offline.

Note:

Smart View recommends a maximum grid size of 65,000 cells.

Note:

Applying a different data source to a sheet containing a data form than the data source originally used to create the data form is not supported.

Note:

Excel worksheets are always protected to avoid entering data for read-only cells. Therefore, some Excel functions, such as AutoSum and F9, are disabled.

When you work with data forms rendered in Excel with Smart View:

• Member names are indented based on their level in the hierarchy. They are also indented if you print the data form to a PDF file.

Note:

Financial Management and Hyperion Enterprise data forms are custom-formatted and do not have indentation.

- For members with aliases, if the administrator selected the Display Alias option, aliases are displayed on the rows, columns, page, and POV.
- Values submitted back to the database from Excel must be non-formatted data.
- If a data form is currently loaded in Excel and the administrator changes the data form definition on the server side, Oracle recommends that you close the data form and reload it. This action ensures that the newest data form definitions are displayed.
- Multiple levels in an outline are displayed differently in Smart View than pages on the Planning Web application. Smart View displays up to four levels, while the Web application displays up to two levels.
- If a Planning administrator hides a dimension in the row axis of a Planning data form, this dimension does not display in the row header of the data form in Smart View.

For information about working with data forms in an Excel worksheet, see the following topics:

- "Opening Data Forms in Excel" on page 59
- "Navigating in Data Forms" on page 59
- "Viewing Data Form Instructions" on page 60
- "Viewing Multiple Data Forms in Excel" on page 60
- "Selecting Members for Financial Management and Hyperion Enterprise Data Forms" on page 60
- "Selecting a Range of Cells" on page 63
- "Saving Custom Formats" on page 63
- "Copying and Pasting Cells" on page 64
- "Entering Percentage Values in Planning" on page 64
- "Entering Date Values in Planning" on page 64

- "Writing #Missing Values" on page 65
- "Adding Cell Text in Data Forms" on page 65
- "Subtotaling Values in Planning" on page 67

Opening Data Forms in Excel

When you open a data form in Excel, the data form displays existing values for the selected members. You can enter new values or change the existing values.

If you open a group of data forms simultaneously, each is opened in a separate worksheet in the current workbook. Enough worksheets are created to accommodate them all.

- To open a data form:
- 1 Open Excel and connect to a data source. See "About Independent Provider Connections" on page 35.
- 2 Select Hyperion. then Forms, then Select Form.

The Select Form dialog box displays the Hyperion Enterprise, Essbase, Financial Management, Planning, and Workforce Planning data forms to which you have access.

- 3 From **Folders**, select the folder that contains the data form you want to open.
- 4 From **Forms**, select the data form or forms you want to open.

Note:

If the members in the data form you open have no data, this message (or one that your administrator has created) displays: "There are no valid rows of data for this data form."

- 5 Optional: To view any instructions associated with the data form, click View Instructions.
- 6 Click OK.

For Hyperion Enterprise, the Member Selection dialog box displays after you open a data form in Excel. To select members for the data form, perform step 3 to step 10 in "Selecting Members for Financial Management and Hyperion Enterprise Data Forms" on page 60. Alternatively, you can click the OK tab in the Member selection dialog box to update the POV drop down lists with the default POV dimension members of the Hyperion Enterprise data source.

Navigating in Data Forms

Use the following methods to navigate a data form:

- Press the arrow keys to move forward, backward, up, or down in a data form.
- Press Tab to move to the next cell in the row. Press Shift+Tab to move to the previous cell.
- Press Enter to move to the next cell in the column. Press Shift+Enter to move to the previous cell.
- To navigate among open data forms, click the worksheet tabs at the bottom of the grid.

Viewing Data Form Instructions

Your administrator may have included information to guide you in preparing data, or to explain the information shown on the data form. For Hyperion Enterprise data forms, the administrator can include the name and descriptions of data forms. You can view the instructions or descriptions associated with a data form.

- To view the instructions for a data form in Excel:
- 1 On the **Select Form** page, select a data form and click **View Instructions** .
 - If the data form is already displayed in Excel, select Forms, then Instructions.
- 2 When you finish reading the text, click **OK** to close the **Instructions** window.

Viewing Multiple Data Forms in Excel

You can view multiple data forms in Excel at the same time.

- To view multiple data forms in an Excel workbook:
- 1 Select the workbook that contains the data forms that you want to view.
- 2 From the Window menu, select New Window.
- 3 Select the data form that you want to view.
- 4 From the Window menu, select Arrange to select how you want the worksheets to be displayed.

Tip:

To view only sheets in the active workbook, select Windows of Active Workbook.

Selecting Members for Financial Management and Hyperion Enterprise Data Forms

You can select the POV dimension members for the data form. After you select members, the POV drop-down lists are updated with your selections.

You use the Member Selection dialog box to select POV dimension members. You can view labels or descriptions for dimension members. For example, when you select to view descriptions, the P_Series member is displayed as "P_Series - Phones and PDAs".

You can also search for POV dimension members. For more information, see step 7 on page 61.

Note:

In Financial Management, if you use the @CUR functionality in a data form, when the form is imported into Smart View, the @CUR member is taken from the background POV for the selected application.

- To select dimension members:
- 1 From any POV drop-down list, click the Ellipsis button (...) to open the **Member Selection** dialog box.
- 2 Select a member to open the **Member Selection** dialog box.
- 3 From the **Dimension** drop-down list, select a dimension.
- 4 Optional: To display active entities only, select Active Members.

Note:

The Active Member option is available only if the application has been set up for Organization by Period. For information on Organization by Period, see the *Hyperion Financial Management – System 9 Administrator's Guide*. The Active Member option is not supported for Hyperion Enterprise data source.

- 5 Highlight a member, then use one of the following methods to select members:
 - To select individual members, select the check box next to each member that you want to use.
 - To select all members for the highlighted member, select the box.
 - To select all immediate children for the highlighted member, select the box, and from the drop-down list, select Children.
 - To select all base members for the highlighted member, select the box, and from the drop-down list, select Base Members.
- 6 Click , to move the selected members to the **Selection** list.

Tip:

To remove members from the Selection list, select the members, then click Remove, . To remove all members from the Selection list, click Remove All,

- 7 Optional: To search for members in the selected dimension, complete the following steps:
 - a. Click and enter the member name or pattern for which you want to search in the text box.

You can enter the start of a text pattern for the search, or you can use a trailing asterisk as a wildcard symbol. For example, to find EastSales, you can enter "east" or "ea*.'

- b. Click to find the first member within the dimension that matches the search criteria.

 Select the check box next to the member if you want to use it and then click.
- c. Click again to search for the next occurrence and, if you want to use the member, select the check box next to the member and click.

Tip:

If you are at the bottom of the member list, click to find the next member that matches the search criteria.

- 8 Optional: To reorder members in the Selection list, click the Up or Down arrows or the Make Lowermost and Make Topmost buttons above the list to move members.
- 9 When you finish selecting members, click **OK**.

Adding Members to Financial Management Data Forms

When administrators define data forms, they can choose to allow form users to insert additional lines of data. If an administrator has enabled this for the form, a red border is displayed around a cell. You can select additional members, and enter and save data for those members. The new entries are inserted into the form and the totals are updated.

For example, a form may have been defined for an account with intercompany transactions for IC1, IC2, and IC4. You can select members IC3 and IC5, and enter and save data for those members. The form is refreshed with the new data and the new lines are displayed in the appropriate hierarchical order.

- To add members to data forms:
- 1 Open a data form.

Tip:

A red border is displayed for cells to which you can add members.

- 2 Right-click and select Hyperion, then select Forms, and Add Member.
- 3 From the member selector, select the members for which to enter data and click **OK.**

Tip

To select multiple members, hold Ctrl and select members. Use Add All and Remove All to add or remove all members.

4 Click OK.

The new members are listed in the member list.

Using Financial Management Linked Forms

When administrators create data forms, they can define links from one form to another to enable drill-through to a more specific data entry view. For example, a form that contains summary account balances can link to a corresponding form with the account details. The link from one form to another applies to an entire row. A data form can contain up to 64 linked forms.

For rows that contain links to forms, the cell text is displayed in red.

- To use linked forms:
- In a data form, select a row that contains linked forms.
- 2 Right-click and select HFM Linked Forms, then select the form name.
 - A new form is displayed in a separate browser window.
- 3 When you finish using the linked form, click Close.

Selecting a Range of Cells

You can select any group of cells in a data form as long as the selection is rectangular and contiguous.

After you select a group of cells, you can copy and paste them, or adjust and spread data values in them. For more information, see "Copying and Pasting Cells" on page 64 and "Adjusting and Spreading Data Values" on page 72.

Saving Custom Formats

Customization of a data form in Excel is preserved only if the customization was done outside of the main grid. Customization to the grid is not preserved except for the thousands and decimals separators.

Note:

In Excel 2002 (XP) and Excel 2003, you change the thousands and decimal separators by selecting Tools, then Options, then International. To specify a new separator, enter a new separator in the Decimal separator or Thousands separator boxes. Additionally, the thousands separator must be enabled and selected from Hyperion, then Options, then Display tab, then Use Thousands Separator.

Searching for a Page in Planning

If the Planning administrator sets up multiple page dimensions for a form, you select the page with the data you require from the page drop-down.

- To search for a page in Planning:
- 1 Click in the page dimension you want to search to highlight it.
- Optional: The page drop-down can contain hundreds of pages. To find a page in a long list, type one or more characters of the page name you want. For example, to search for USD in the currency page, type U.
 - If the characters you type match the name of a page, the full name of the page is displayed in the page drop-down.
- 3 Select the page name containing the data with which you want to work.

Copying and Pasting Cells

In Excel, you can copy data values within a data form, and from one data form to another. In a single copy and paste operation, you can copy values from one cell to another cell, from one cell to many cells, or from many cells to many cells.

Note:

If you copy and paste a value with supporting detail in the main grid, only the value is copied and pasted, and not the supporting detail.

- To copy and paste data:
- 1 Select the worksheet in which you are copying and pasting cells.

Tip:

If the worksheet is protected, select Tools, then Protection, then Unprotected Sheet.

2 Select the cell or group of cells that contain the data that you want to copy.

For information on selecting multiple cells, see "Selecting a Range of Cells" on page 63.

3 Right-click and select Copy to copy the values in the selected cells to the clipboard.

To copy data in other applications, use that application's Copy command.

- 4 Select the cell or group of cells to which you want to paste the data.
- 5 Click Paste.

Note:

If you try to copy and paste using Ctrl+C and Ctrl+V twice, the value is not copied the second time.

Entering Percentage Values in Planning

If a Planning administrator has set the data type for a member to Percentage, the member is displayed with a percent sign (%) in its cell. When you change a percentage value, Smart View accepts the new value and displays a percent sign.

Entering Date Values in Planning

If a Planning administrator sets Date as a member's data type, the member is displayed as a date. The default date format is DD/MM/YY. When you change a date, Smart View accepts the new value and displays it. If a member's data type is not Date, you cannot successfully enter a date in the cell.

Writing #Missing Values

A cell that displays #Missing has no data value in the data form. You can enter a data value by selecting the cell and entering the value. You can also eliminate irrelevant data in a cell by selecting the cell with the irrelevant data and replacing the data with #Missing.

You can submit a data form to write #Missing values to the database. If you replace #missing with a character (for example, a dash (-)), you can place a dash in the cell and submit that to the database, too. After you select the cells that you want to contain #Missing, press the Delete key or enter #Missing. The cells are set to #Missing when you save the data form.

Note:

#Missing is different from zero. Zero is a data value, and #Missing indicates lack of a data value for the cell. #Missing decreases the size of the database and positively impacts system performance. On data forms that are designed to suppress missing data, rows that contain cells with #Missing values are not displayed after the data form is reloaded.

Note:

You can choose to have Smart View display numeric zeroes instead of #Missing or other text. See "Replacement Labels" on page 212.

Adding Cell Text in Data Forms

You can add annotations called cell text to the cell at any level. You can add cell text at the summary time period level and across multiple dimensions at any level. You can also add cell text for non-level zero members (bottom-up versions), calculated cells (Dynamic Calc), and read-only cells, such as explanations for data analysis of variances and rolling forecasts.

When you work with cell text, keep in mind that you can also use supporting or line item detail to add comments to data. You can use supporting detail to build and communicate bottom-up values, such as travel, for which you need to calculate aggregate values. Cells that contain text are indicated by a dark blue border around the cell.

- To add cell text:
- 1 Open a data form, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 In the data form, select a cell or a range of contiguous cells.
- 3 Select Hyperion, then Cell Text. For Office 2007, click Cell Text in the Task section of the Hyperion ribbon.
- 4 In the **Cell Text** window, enter the text that you want to add.
- 5 Click Save.

Viewing and Editing Cell Text in Data Forms

You can use the Cell Text window to view cell text for a single cell or for a range of contiguous cells. You cannot view cell text directly in a data form.

- To view or edit cell text:
- 1 Open a data form, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 In the data form, select a cell or a range of cells that contain cell text.
 - When a cell contains cell text, an indicator is displayed in the cell.
- 3 Select Hyperion, then Cell Text. For Office 2007, click Cell Text in the Task section of the Hyperion ribbon.
- 4 In the **Cell Text** window, view or edit the text.
- 5 When you finish viewing or editing cell text, click Save.

Submitting Cell Text in Planning

After you enter or edit cell text in a Planning form, you can submit it to the Planning database.

- To submit cell text to the Planning database:
- 1 Open the data form, if necessary, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 Edit the cell text.
- 3 Select Hyperion, then Submit Data to save the text to the database. For Office 2007, click Submit in the Review section of the Hyperion ribbon.

A blue triangle in a cell's upper right corner indicates that it contains cell text.

Working with URLs Linked to Cells

Data Sources: Planning

Planning data cells can be linked to URLs, and you can launch a URL for any data cell from Smart View.

Adding, Editing, and Deleting Cell URLs

- To add, edit, or delete a URL linked to a cell:
- 1 On a data form, select a data cell.
- 2 Select Hyperion, then Cell Text.
- 3 Click the URL tab.
- 4 In the text box, enter, modify, or delete the URL as necessary.

Launching Cell URLs

- To launch a URL for a data cell in a new browser:
- 1 Select the cell that contains the URL to launch.
- 2 Select Hyperion, then Document Link.

The URL that is linked to the cell opens in a new browser.

Subtotaling Values in Planning

The following points explain how values are subtotaled and totaled in data forms:

- Dimension member subtotals are calculated based on factors such as the hierarchies and logic of the Essbase outline and the member properties an administrator sets.
- If the Calculate Data Form calc script is selected to launch during a save operation, all subtotals in the data form are recalculated based on their members' aggregation properties and the design and layout of the data form.
- When data is saved, Essbase automatically calculates members that are set to dynamically calculate, so the data form does not require a calc script to calculate these members. This Essbase calculation excludes level 0 members.
- Calculations are based on stored values, which are not necessarily the same as displayed
 values. For example, the values that you see in the data form may be based on scaling or on
 precision settings.
- Only members displayed on the data form are calculated.
- If you have read but not write access to a member, subtotals correctly include its value even though it is read-only.
- If a Calculate Data Form business rule is associated with the data form, subtotals are calculated automatically when the script is launched. See "Executing the Calculate Data Form and Calculate Currencies Business Rules" on page 72 for more information.

Changing Currency in Planning Data Cells

If a Planning administrator enabled the functionality, you can enter data in a currency other than a cell's base currency. Currencies in the drop-down list can be designated as the local currency.

Note:

To override the base currency for an entity, the cell must be displayed in the local currency, and its version must be bottom-up. The application must be a multi-currency application and the data form should support multi-currency.

- To enter cell data in a local currency other than the base currency for the cell:
- 1 In a data form, select a local currency member for the cell.
- 2 Optional: To look up the currency's code, select View, then Currency.

Available Currencies shows the application's currencies. Note the Currency Code for the currency you want to work with, and close the window. If you cannot select View, then Currency, multiple currencies are not enabled for this application or data form.

- 3 In the right column, HSP_InputCurrency, type the new Currency Code in the data cell.
 - Typing the currency code in the data cell overrides the base currency for the entity.
- 4 Click **Submit** to submit the new currency code to the Planning server.
- 5 Enter the currency value in the left column, HSP_InputValue, of the data cell.
- 6 Click **Rules on Form** and select the Calculate Currencies rule to calculate and save the new currency value.

If the Calculate Currencies calc script is set to run when the data form is saved, and the data form is enabled for multiple currencies, the data value is displayed in the currency you selected.

Submitting Data

You can submit data to the data source in either connected or disconnected mode. Submitting data in offline mode is not supported for Hyperion Enterprise.

In valid ad hoc and data form grids in Excel, you can modify and save data (any type) when Smart View is disconnected from the data source and then submit your modified data to the data source when you reconnect.

If you modify **member names**, however, you trigger free-form mode and must refresh the data before submitting it. Because refreshing replaces data in the worksheet with data from the data source, your modifications are lost before you submit them.

Note:

Smart View must be installed on the client computer; otherwise, you lose your changes when you submit the data. In addition, there must not have been changes to the form, user privileges, or other settings on the server.

The following rules apply when data is submitted:

- All dimensions must appear in the same axis as they did before the data was synchronized. For example, if accounts were on the row when you loaded the data form into Smart View, then they must be on the row on the server when the you submit the data form. If not, you receive an error message. This is true for all axes (rows, columns, pages and POVs).
- All pages must appear in the same order as when the data form was loaded. If not, you receive
 an error message.

- You can change the order of the dimensions in the row or column. For example, if the data form had Entities and Accounts on the row, and you reverse this order and submit the data form, when you save the data form, all the data is written to the correct members.
- When using Financial Management or Hyperion Enterprise forms, you can lock any cell or range of cells to apply cell protection until the data is refreshed or submitted. Note that locking the cell does not lock the actual data cube in Financial Management, but only the cell in the form. When the data is refreshed or submitted, the cell is no longer locked.

If all of these rules are followed, the grid is saved. In some cases, some of the cells submitted may no longer exist on the data form. This can occur when the data form definition changes, when rows/columns are suppressed, or when access privileges change. In these cases, only the cells that are writable and exist on the new data form definition are saved. However, you will see a message that some of the cells were not saved.

This applies to both cells and supporting detail changes, and also applies to both online and offline modes.

To submit data to the data source, select **Hyperion**, then **Submit Data**. For Office 2007, click **Submit** in the **Review** section of the **Hyperion** ribbon.

Working with Business Rules for Planning

You can launch business rules and calculation scripts that recalculate data in Essbase. The relevant data form in Excel is updated with the new data. The calculation scripts you can access are shown by plan type.

When you launch a business rule, the rule can display a runtime prompt, asking you to enter information.

If a calculation is successful, the values in the Essbase database reflect the results.

For more information about business rules, see:

- "Launching Business Rules in Excel" on page 69
- "Launching Business Rules From a Data Form" on page 70
- "Entering Runtime Prompts for Business Rules" on page 70
- "Executing the Calculate Data Form and Calculate Currencies Business Rules" on page 72

Launching Business Rules in Excel

You can launch business rules that recalculate data in Essbase.

- To launch a business rule in Excel:
- 1 Open a data form, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 Save any unsaved data in the spreadsheet.

Any data not saved on the spreadsheet is lost when you launch a business rule.

3 Select Hyperion, then Calculation Options, then Business Rules.

The Business Rules dialog box is displayed.

- 4 In the Plan Type frame, select the plan type associated with the rule you want to use.
- 5 Select a rule from the rules listed for that plan type, then click **Launch**.

If the business rule contains runtime prompts, complete step 2 of "Entering Runtime Prompts for Business Rules" on page 70.

If the calculation is successful, the values in the Essbase database reflect the results of the calculation.

- 6 Click Close.
- 7 To view the new values in the data form, select **Hyperion**, then **Refresh**. For Office 2007, click **Refresh** in the **Review** section of the **Hyperion** ribbon.

Launching Business Rules From a Data Form

Sometimes one or more business rules are associated with a data form.

- To launch a business rule associated with a data form:
- 1 Open a data form, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 Save any unsaved data in the spreadsheet.

Any unsaved data on the spreadsheet is lost when you launch a business rule.

- 3 Select Hyperion, then Calculation Options, then Rules on Form.
- 4 Select a rule from the rules listed, then click **Launch**.

If the business rule contains runtime prompts, complete step 2 of "Entering Runtime Prompts for Business Rules" on page 70.

If the calculation is successful, the values in the Essbase database reflect the results of the calculation.

5 To view the new values in the data form, select **Hyperion**, then **Refresh**. For Office 2007, click **Refresh** in the **Review** section of the **Hyperion** ribbon.

Entering Runtime Prompts for Business Rules

When you launch a business rule or save a data form that launches a business rule as a property of the data form, the rule can issue a runtime prompt, which prompts you to enter information. When you work with runtime prompts, keep the following points in mind:

• If a business rule has a runtime prompt and Use Members on Data Form is selected by an administrator in Planning, the default member on the runtime prompt window matches the current member in the page and/or the POV selection of the open data form.

- Members on the Member Selection page are filtered by the current user's security definitions and by any limitations set for the runtime prompt (for example, only Descendants of Q1). You cannot select a shared member in a runtime prompt.
- You can launch Essbase calculation scripts from Smart View, but you cannot launch graphical business rules and enhanced calculation scripts.
- To enter a runtime prompt for a business rule:
- 1 Launch a business rule that has a runtime prompt, using the procedure in "Launching Business Rules in Excel" on page 69.

If you select Submit Data on a data form that has been designed to launch a business rule when saved, the Runtime Prompts dialog box is displayed automatically, if necessary.

2 In the Runtime Prompts dialog box, enter the necessary information.

The icon in front of the prompt indicates the type of expected input, as shown in the following table:

| Icon | Expected Type of Input |
|------|--|
| ** | Single member selection |
| | Multiple member selections |
| 123 | Numeric value |
| ABC | Text value |
| | Note: Text value is for use only with enhanced calc scripts. String type cannot be used with Graphical Scripts. |
| | Dimension from the database Note: Dimension is for use only with enhanced calc scripts. Dimension type cannot be used with Graphical Scripts. |

3 Click Launch.

If the calculation is successful, the values in the Essbase database reflect the results of the calculation.

4 To view the new values in the data form, select **Hyperion**, then **Refresh**. For Office 2007, click **Refresh** in the **Review** section of the **Hyperion** ribbon.

Executing the Calculate Data Form and Calculate Currencies Business Rules

Use the Rules on Form command to select and execute the Calculate Data Form and Calculate Currencies business rules.

The Calculate Data Form business rule is created for each data form to calculate subtotals. The Calculate Currencies business rule is created for data forms that include multiple currencies in a row, column, or page to enable the conversion of values among available currencies.

The order in which business rules are launched is very important and may affect the data. If you plan to launch both Calculate Data Form and the Calculate Currencies business rules, it is important that you run the conversions first, before subtotaling the data form.

- To launch the Calculate Data Form and Calculate Currencies business rules in Excel:
- ${\bf 1} \quad {\bf Open \ a \ data \ form, \ using \ the \ procedure \ described \ in \ "Opening \ Data \ Forms \ in \ Excel" \ on \ page \ 59.}$

Any data that is not saved on the spreadsheet is lost when you launch the business rule.

2 Select Hyperion, then Calculation Options, then Rules on Form.

The business rules associated with the data form are displayed in the Business Rules dialog box.

- 3 Complete one or both of the following actions:
 - To convert currencies, select Calculate Currencies.
 - To calculate subtotals, select Calculate Data Forms.
- 4 Click Launch.

If the calculation is successful, the values in the Essbase database reflect the results of the calculation.

Adjusting and Spreading Data Values

You can make adjustments to data values in a data form using the following methods:

- Adjust values by increasing or decreasing by a numeric value or percentage
- Spread values over a range of cells

You cannot adjust a data value if the member dimension is read-only or if the value is null (that is, if the value is #Missing).

For instructions on adjusting data values and spreading, see the following topics:

- "Adjusting Values" on page 73
- "Spreading Data for Time Periods" on page 73
- "How Spreading Data Works" on page 74
- "Spreading Data with Cell Locking in Planning" on page 77
- "Examples of Spreading Data with Cell Locking" on page 78

Note:

In the Options dialog box, Display tab, if you specified a non-numeric symbol for missing values (for example, a hyphen) in the #NoData/Missing Label text box, you cannot use the Adjust Data feature.

Adjusting Values

Data Sources: Planning, Hyperion Enterprise, Financial Management

You can adjust the value of one or more cells in a data form by a specified number or percentage if the cells contain numerical data and are not read-only.

- ➤ To adjust data values:
- 1 Open a data form, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 In the data form, click the data cell that contains the value to adjust.
- 3 Select Hyperion, then Adjust. For Office 2007, click Adjust in the Task section of the Hyperion ribbon.
- 4 From Adjust Data, select an option then enter the number or percentage by which you want to adjust the value of the cell:
 - Increase Selected Cells by fixed Percentage
 - Decrease Selected Cells by fixed Percentage
 - Add Fixed Value to Selected Cells
 - Subtract Fixed Value from Selected Cells
 - Multiply Selected Cells by a Fixed Value
 - Divide Selected Cells by a Fixed Value
- 5 Click Adjust Data. For information on how adjusting data can affect other cells, see "How Spreading Data Works" on page 74

Spreading Data for Time Periods

Data sources: Planning

Working with a data form page in Excel, you can spread, or distribute, values in several ways:

- Spread the value of a summary time period to its base time periods or to the first parent or first child of the parent time period
- Spread values among children and parents proportionally, based on existing distribution
- Spread values based on the weekly distribution of a quarter, which could be 4-4-5, 5-4-4, 4-5-4, or None (as set up by the budget administrator)
- Temporarily lock the values of certain cells while spreading data over time periods

Note:

You cannot spread data in a summary time period that includes members with mixed currency types.

- To spread data for time periods:
- 1 Open a data form, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 Enter the new value in the selected cell.

The value is distributed according to the rules in the table on "How Spreading Data Works" on page 74.

3 Click Save.

How Spreading Data Works

Factors like account type, existing distribution, member hierarchies, and data type affect how values are distributed. Whether or not a cell is locked also affects data distribution. For information on spreading data with locked cells, see "Spreading Data with Cell Locking in Planning" on page 77.

The following examples show the results if you enter or change a currency or non-currency value.

| Time Balance Property of the Account | New Value Distribution | Examples |
|--|---|---|
| FLOW Revenue, Expense, Saved Assumption (where the Time Balance property is set to Flow) | To all children and parents proportionally, based on the existing distribution. The new value affects the entire summary period roll ups hierarchy so that the parent time period is always the sum of its children. If there is no existing distribution (that is, if the values for all children are zeros or #Missing) and the changed value is a quarter, the new value spreads down proportionally, based on the weekly distribution, which can be 4-4-5, 4-5-4, 5-4-4, or evenly if the account's spreading is set to None). If the changed parent is a year Total or some other kind of summary time period, the value is spread evenly. | Example 1 You change Qtr 1 from 250 to 500. Before the change, the months of Qtr 1 have the following values: Jan = 100 Feb = 50 Mar = 100 Result: The 500 is distributed to the childrer of Qtr 1 proportionally, replacing previous values as follows: Jan = 200 Feb = 100 Mar = 200 The 500 is aggregated to the parents of Qt 1. If Year Total was formerly 1000, its new value is 1250. Example 2 You change March from 100 to 200. Result: March, Qtr 1, and Year Total all increment by 100. Jan and Feb remain unchanged. |

| Time Balance Property of the Account | New Value Distribution | Examples |
|--|--|--|
| FIRST All types of Accounts | Upward to its first parent and downward to its | Example 1 |
| | child only if the changed cell is the first child of its parent time period. | You change Qtr 1 from 20 to 40. Before the change, the months of Qtr 1 have the |
| | The summary time period is always equal to the first of its child time periods. | following values: Jan = 20 |
| | If there is no existing distribution (that is, | Feb =15 |
| | values for all the children are zeros or are missing), the new value is copied to each of | Mar = 05 |
| | the children. | 01 = 20 |
| | | Result : The 40 is distributed to the children of Qtr 1 proportionally, replacing their previous values as follows: |
| | | Jan = 40 |
| | | Feb = 15 |
| | | Mar = 05 |
| | | Q1 = 40 |
| BALANCE Asset, Liability, Equity, Saved | Downward to the last child and upward to the | Example 1 |
| Assumption (where the Time Balance property is set to Balance) | parent only if the changed cell is the last child of its parent time period. | You change Qtr 1 from 30 to 50. |
| property is commonly | The summary time period is always equal to the last of its child time periods. If there is no existing distribution (that is, if the values for all children are zero or missing), the new value is spread across the children. | Result : March also changes to 50. Jan and Feb do not change. Year Total does not change because Qtr 1 is not the last child of |
| | | Year Total. |
| | | Example 2 |
| | | You change Qtr 4 from 100 to 50. |
| | | Result : Dec changes to 50 because it is the last child of Qtr 4. Oct and Nov remain unchanged, as do quarters 1, 2, and 3. Year Total changes to 50 because Qtr 4 is the last child of Year Total. |
| | | Example 3 |
| | | You change Qtr 2 to 100. Before the change, the months of Qtr 2 have the following values: |
| | | Apr = 0 |
| | | May = 0 |
| | | June = 0 |
| | | Result: |
| | | Apr = 100 |
| | | May = 100 |
| | | June = 100 |
| | | Year Total is unchanged. |

| ASSUMPTION, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_365 Revenue, Expense, Saved Assumptio | Time Balance Property of the Account | New Value Distribution | Examples |
|--|---|---|--|
| distribution. The new value affects the entire summary time period of lups hierarchy that the parent is always the average of its children. Assumes an equal number of days in each period, such as 30 days for each month. Assumes an equal number of days in each period, such as 30 days for each month. AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarter must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. MMLswifing is treated as 0 in the unmerator, and all of the days are included in missing months in the denominator. Thus, for example, QIR means three months, not YID. • All months are included in the calculation. MMLswifing is treated as 0 in the unmerator, and all of the days are included in missing months in the denominator. Thus, for example, QIR means three months, not YID. • All months are included in the calculation. MMLswifing is treated as 0 in the unmerator, and all of the days are included in missing months in the denominator. Thus, for example, QIR means three months, not YID. • All months are included in the calculation, in the denominator. Thus, for example, QIR means three months, not YID. • All months are included in the calculation, in the denominator. Thus, for example, QIR means the days in question of days in 10 signs of days in 10 sig | AVERAGE Revenue, Expense, Saved | | Example 1 |
| children. Assumes an equal number of days in each period, such as 30 days for each month. Assumes an equal number of days in each period, such as 30 days for each month. AWG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years, February assumed to have 28 days, and Qtr 1 is assumed to have 29 days. Value Entered Number of Days in 2 year, in which February has 29 days and Qtr 1 is assumed to have 29 days. Weighted daily average based on the actual number of days in Qt, the result is as follows: (9,000 * 31 * 8,000 * 28 * 8,000 * 31) / 90 - 8,344 AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in Qt, the result is as follows: (9,000 * 31 * 8,000 * 28 * 8,000 * 31) / 90 - 8,344 For a leap year, you enter values for Jan, Feb, and Mar. February is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days. | Assumption, (where the Time Balance property is set to Average) | distribution. The new value affects the entire summary time period roll ups hierarchy so | change, the months of Qtr 1 have the |
| period, such as 30 days for each month. Mar = 00 Q1 = 05 Result: Jan = 10 Feb = 20 Mar = 00 Q1 = 10 Example 1 Vou each verage) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. ##xi #saing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means there months, not QTD. • All months, not QTD. • All months, not QTD. • All months, not YDD. • All months, not YDD. • All months, not YDD. • All weighted daily average based on the actual number of days in Q1. Using 28 for the number of days in Q1. | | - | Jan = 05 |
| Mar = 00 Q1 = 05 Result: Jan = 10 Feb = 20 Mar = 00 Q1 = 10 Example 1 You enter values for Jan, Feb, and Mar. For any year, including leap years, February is assumed to have 28 days, and Qtr 1 is assumed to have 28 days, and Qtr 1 is assumed to have 28 days, and Qtr 1 is assumed to have 28 days and Qtr 1 is assumed to have 29 days. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in 21, the result is as follows: (9,000 *31 / 90 - 8,344 Example 1 For a leap year, you enter values for Jan, Feb, and Mar. Feb Jan, You cannot customize month labels, although you can use aliases. | | | Feb = 10 |
| Result: Jan = 10 Feb = 20 Mar = 00 Q1 = 10 AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not QTD. **Result** Q1 = 8,344 **The average for Qtr 1 is calculated as follows: Q1 (1) Multiply the va | | period, such as 30 days for each month. | Mar = 00 |
| AVG_ACTUAL Revenue, Expense, Saved ASSumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. *#insair.gr is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. Value Entered Number of Days Har = 10 Example 1 You enter values for Jan, Feb, and Mar. February laways has 28 days. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. | | | Q1 = 05 |
| AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: You cannot customize month labels, although you can use aliases. Years must have twelve months, and quarters must be the sum of three base months. You cannot change the fiscal start month after the application is set up. All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. ANG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: Weighted daily average based on 365 days in a year, assuming to have 28 days, and Qtr 1 is assumed to have 28 days, and Qtr 1 is assumed to have 28 days, and Qtr 1 is assumed to have 29 days. As you work with AVG_ACTUAL, keep these points in mind: Weighted daily average based on the actual number of days in 12 set. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: Weighted daily average based on the actual number of days in a year. This accounts for leap years, not Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days. As you work with AVG_ACTUAL, keep these | | | Result: |
| AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD. You are means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, including leap years, February is assumed to have 29 days. As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. | | | Jan = 10 |
| AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #Missing is treated as 0 in the numerator, and all off the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in Q1. Using 28 for the number of days in Q1. Using 28 for | | | Feb = 20 |
| AVG_365 Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on 365 days in a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in Q1, the result is as follows: 0,000 *31 *8,000 *28 *8,000 *31) / 90 = 8,344 Example 1 You enter values for Jan, Feb, and Mar, For any year, including leap years, February is assumed to have 28 days, and Qtr 1 is assumed to have 90 days. Value Entered Number of Days Jan = 9,000 31 Feb = 8,000 28 Mar = 8,000 31 Q1 90 (the total days for Jan, Feb, and Mar) Result: Q1 = 8,344 The average for Qtr 1 is calculated as follows: (1,1) Multiply the value for each month in Qtr 1 by the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Iterative as the number of days in Q1. Iterative as the number of days in Q1. Iterative as the number of days in Q1. Using 28 for the number of days in Q1. Iterative as the | | | Mar = 00 |
| Assumption, (where the Time Balance property is set to Average) and a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #\ffloat is signing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. ANG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) in a year, assuming that February always has 28 days. This does not account for leap years, including leap years, February is assumed to have 28 days, and Qt 1 is assumed to have 90 days. Value Entered Number of Days All months are included in the calculation. #\ffloat is assumed to have 28 days, and Qtr 1 is assumed to have 90 days. Value Entered Number of Days Mar = 8,000 31 Qt 90 (the total days for Jan, Feb, and Mar. For any year, including leap years, February is assumed to have 90 days. Value Entered Number of Days Weighted daily average based on the actual number of days in 1 that month, (2) sum the values, and (3) divide the total days for Jan, Feb, and Mar. February is assumed to have 91 days. Weighted daily average based on the actual number of days in 19, the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = 8,344 AVG_ACTUAL Revenue, Expense, Saved As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. | | | Q1 = 10 |
| Assumption, (where the Time Balance property is set to Average) and a year, assuming that February always has 28 days. This does not account for leap years. As you work with AVG_365, keep these points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #\ffloat is signing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. ANG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) in a year, assuming that February always has 28 days. This does not account for leap years, including leap years, February is assumed to have 28 days, and Qt 1 is assumed to have 90 days. Value Entered Number of Days All months are included in the calculation. #\ffloat is assumed to have 28 days, and Qtr 1 is assumed to have 90 days. Value Entered Number of Days Mar = 8,000 31 Qt 90 (the total days for Jan, Feb, and Mar. For any year, including leap years, February is assumed to have 90 days. Value Entered Number of Days Weighted daily average based on the actual number of days in 1 that month, (2) sum the values, and (3) divide the total days for Jan, Feb, and Mar. February is assumed to have 91 days. Weighted daily average based on the actual number of days in 19, the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = 8,344 AVG_ACTUAL Revenue, Expense, Saved As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. | AVG 365 Revenue, Expense, Saved | Weighted daily average based on 365 days | Example 1 |
| points in mind: • You cannot customize month labels, although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. Value Entered Number of Days Jan = 9,000 31 Feb = 8,000 28 Mar = 8,000 31 Q1 90 (the total days for Jan, Feb, and Mar) Result: Q1 = 8,344 The average for Qtr 1 is calculated as follows: (1) Multiply the value for each month in Qtr1 by the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Path and Path | Assumption, (where the Time Balance property is set to Average) | in a year, assuming that February always has 28 days. This does not account for leap years. | You enter values for Jan, Feb, and Mar. For any year, including leap years, February is |
| although you can use aliases. • Years must have twelve months, and quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. **Result:* Q1 = 8,344 The average for Qtr 1 is calculated as follows: (1) Multiply the value for each month in Qtr1 by the number of days in that month, (2) sum the values, and (3) divide the total by the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Peb and 90 forthe number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Peb and 90 forthe number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Peb and 90 forthe number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using 28 for the number of days in Q1. Using | | | |
| Years must have twelve months, and quarters must be the sum of three base months. You cannot change the fiscal start month after the application is set up. All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: You cannot customize month labels, although you can use aliases. | | You cannot customize month labels, | Value Entered Number of Days |
| quarters must be the sum of three base months. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. Feb = 8,000 28 Mar = 8,000 31 Q1 90 (the total days for Jan, Feb, and Mar) Result: Q1 = 8,344 The average for Qtr 1 is calculated as follows: (1) Multiply the value for each month in Qtr1 by the number of days in that month, (2) sum the values, and (3) divide the total by the number of days in Q1. Using 28 for the number of days in Peb and 90 for the number of days in Q1. the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = 8,344 Feb = 8,000 31 | | although you can use aliases. | Jan = 9,000 31 |
| wonths. • You cannot change the fiscal start month after the application is set up. • All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. Mar = 8,000 31 Q1 90 (the total days for Jan, Feb, and Mar) Result: Q1 = 8,344 The average for Qtr 1 is calculated as follows: (1) Multiply the value for each month in Qtr by the number of days in Q1. Using 28 for the number of days in Q1, the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = 8,344 Example 1 For a leap year, you enter values for Jan, Feb, and Mar. February is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days, and Qtr 1 is assumed to have 29 days. Value Entered Number of Days Jan = 9,000 31 | | quarters must be the sum of three base | Feb = 8,000 28 |
| after the application is set up. All months are included in the calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: You cannot customize month labels, although you can use aliases. Mar) Result: Q1 = 8,344 The average for Qtr 1 is calculated as follows: (1) Multiply the value for each month in Qtr1 by the number of days in the number of days in Feb and 90 for the number of days in Q1, the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = 8,344 Example 1 For a leap year, you enter values for Jan, Feb, and Mar. February is assumed to have 29 days, and Qtr 1 is assumed to have 91 days. Value Entered Number of Days Jan = 9,000 31 | | | Mar = 8,000 31 |
| calculation. #Missing is treated as 0 in the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: You cannot customize month labels, although you can use aliases. Value Entered Number of Days Jan = 9,000 31 | | | Q1 90 (the total days for Jan, Feb, and Mar) |
| the numerator, and all of the days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. AVG_ACTUAL Revenue, Expense, Saved Assumption, (where the Time Balance property is set to Average) Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: You cannot customize month labels, although you can use aliases. Q1 = 8,344 The average for Qtr 1 is calculated as follows: (1) Multiply the value for each month in Qtr1 by the number of days in Q1. Using 28 for the number of days in Q1, the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = 8,344 Example 1 For a leap year, you enter values for Jan, Feb, and Mar. February is assumed to have 29 days, and Qtr 1 is assumed to have 91 days. Value Entered Number of Days Jan = 9,000 31 | | | Result: |
| denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. Year means all twelve months, not YTD. Weighted daily average based on the actual number of days in Peb and 90 for the number of days in Q1, the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = 8,344 Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: Weighted daily average based on the actual number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: You cannot customize month labels, although you can use aliases. Value Entered Number of Days Jan = 9,000 31 | | | Q1 = 8,344 |
| Assumption, (where the Time Balance property is set to Average) number of days in a year. This accounts for leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: • You cannot customize month labels, although you can use aliases. To a leap year, you enter values for Jan, Feb, and Mar. February is assumed to have 29 days, and Qtr 1 is assumed to have 91 days. Value Entered Number of Days Jan = 9,000 31 | | denominator. Thus, for example, QTR means three months, not QTD, and Total | number of days in Q1. Using 28 for the number of days in Feb and 90 for the number of days in Q1, the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = |
| property is set to Average) leap years, in which February has 29 days. As you work with AVG_ACTUAL, keep these points in mind: Yalue Entered Number of Days You cannot customize month labels, although you can use aliases. Jan = 9,000 31 | AVG_ACTUAL Revenue, Expense, Saved | | Example 1 |
| points in mind: Value Entered Number of Days Value Entered Number of Days Jan = 9,000 31 | property is set to Average) | leap years, in which February has 29 days. | |
| You cannot customize month labels, although you can use aliases. Jan = 9,000 31 | | | |
| aithough you can use allases. | | You cannot customize month labels, | |
| | | although you can use aliases. | Jan = 9,000 31 Feb = 8,000 29 |

| Time Balance Property of the Account | New Value Distribution | Examples |
|--------------------------------------|--|--|
| | Years must have twelve months, and | Mar = 8,000 31 |
| | quarters must be the sum of three base months. | Q1 91 (the total days for Jan, Feb, and Mar) |
| | You cannot change the fiscal start month after the application is set up. | Result: |
| | All months are included in the | Q1 = 8,341 |
| | calculation. #Missing is treated as 0 in the numerator, and all days are included in missing months in the denominator. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. Thus, for example, QTR means three months, not QTD, and Total Year means all twelve months, not YTD. | The average for Qtr is calculated as follows (1) Multiply the value for each month in Qt 1 by the number of days in that month, (2) sum these values, and (3) divide the total by the number of days in Q1. Using 29 for the number of days in Feb and 91 for the numbe of days in Q1, the result is as follows: (9,000 * 31 + 8,000 * 29 + 8,000 * 31) / 91 = 8,341 |
| | | Example 2 |
| | | For a non-leap year, you enter values for Jan Feb, and Mar. February is assumed to have 28 days, and Qtr1 is assumed to have 90 days. |
| | | Value Entered Number of Days |
| | | Jan = 9,000 31 |
| | | Feb = 8,000 28 |
| | | Mar = 8,300 31 |
| | | Q1 90 (the total days for Jan, Feb, and Mar) |
| | | Result: |
| | | Q1 = 8,344 |
| | | Using 28 for the number of days in Feb, and 90 for the number of days in Q1, the result is as follows: (9,000 * 31 + 8,000 * 28 + 8,000 * 31) / 90 = 8,344 |

Spreading Data with Cell Locking in Planning

When spreading data over time periods, you can temporarily lock the values of one or more cells to preserve their values when other values are recalculated. You can spread data across time periods based on various calculations and visually review the changes before committing them to the database. This breakback capability is useful when you have seeded your data, and you want to manipulate and analyze values in specific accounts to produce desired results. To see examples of how cell locking works with spreading, see "Examples of Spreading Data with Cell Locking" on page 78.

- ➤ To temporarily lock values:
- 1 Open a data form, using the procedure described in "Opening Data Forms in Excel" on page 59.

- 2 In the data form, select the cell or group of cells that you want to lock.
- 3 Select Hyperion, then Forms, then Lock. For Office 2007, click Lock in the Forms section of the Hyperion ribbon.

A color change indicates that a cell is locked. You can now spread or manipulate data in the other cells however you want, without affecting the locked cells. (For information on how Planning spreads data, see "Spreading Data for Time Periods" on page 73 and "How Spreading Data Works" on page 74.).

4 To unlock a cell, select **Hyperion**, then **Refresh**. For Office 2007, click **Refresh** in the **Review** section of the **Hyperion** ribbon.

Examples of Spreading Data with Cell Locking

Example 1

Account A has the following values before locking and spreading:

| | Jan | Feb | Mar | Q1 |
|-----------|-----|-----|-----|-----|
| Account A | 100 | 100 | 100 | 300 |

In this example, you lock the Feb and Mar values at 100. You then change the value in Q1 from 300 to 600. Because Jan, Feb, and Mar must now total 600 and Feb and Mar are locked at 100 each, Planning calculates Jan to be 400 and fills in that value.

This is how the data is displayed after locking and spreading:

| | Jan | Feb | Mar | Q1 |
|-----------|-----|-----|-----|-----|
| Account A | 400 | 100 | 100 | 600 |

Example 2

Account B has the following values before locking and spreading:

| | Q1 | Q2 | Q3 | Q4 | Year Total |
|-----------|-----|-----|-----|-----|------------|
| Account B | 100 | 100 | 100 | 100 | 400 |

In this example, you lock Q1 and Q2 values at 100 each. You then change Year Total from 400 to 800. Because the yearly total must equal 800 and Q1 and Q2 are locked at 100 each, Planning calculates Q3 and Q4 to be 300 each and fills in those values.

This is how the data is displayed after locking and spreading:

| | Q1 | Q2 | Q3 | Q4 | YearTotal |
|-----------|-----|-----|-----|-----|-----------|
| Account B | 100 | 100 | 300 | 300 | 800 |

Working With Supporting Detail

Planning's Supporting detail feature helps you build and communicate bottom-up values when planning corporate expenses, such as travel, salary, and projects, for which you need to calculate aggregate values. Supporting detail serves as a built-in calculator for developing data that is not in the member outline. (Financial Management supports line item detail, but not supporting detail.) Supporting detail can include text, values, and operators that define how data aggregates.

Supporting detail also provides a way to drill down into data to better understand its basis. For example, if the bottom-level member in your outline is pens, you can add line items in supporting detail for ballpoint, fountain, marker, and so on. Then you can aggregate the detail values to the pen member in the outline.

When working with supporting detail, keep the following points in mind:

- Supporting detail does not change members in the Essbase outline structure.
- To enter, create, change, or delete supporting detail, you must have write access to cells. To view supporting detail, you must have read access.
- You cannot enter, adjust, spread, or save data into aggregate values that have supporting detail. Aggregate values with supporting detail are read-only.
- You can add supporting detail to both target and bottom-up versions.
- You cannot add supporting detail to summary time periods, such as quarters. You can add supporting detail only to base time periods (level zero members).
- Number and precision formatting is not reflected in the Supporting Detail window.
- The sequence of operators in supporting detail follows the logic that Essbase uses to process multiple operators in a complex calculation.
- If numeric values are not defined for one or more cells, you can delete the undefined values from Essbase. If you select to delete the undefined values, #Missing shows in the cells. If you select not to delete undefined values, the values are left as is.
- If you copy and paste a value with supporting detail in the main grid, only the value is copied and pasted, not the supporting detail.
- You can select multiple cells when adding supporting detail in online mode only. In offline mode, you must add supporting detail cell by cell.

For more information about supporting detail, see:

- "Adding Supporting Detail" on page 80
- "Totaling When Supporting Detail Cells Are Blank" on page 80
- "Working with Supporting Detail Hierarchy" on page 81
- "Viewing or Changing Supporting Detail" on page 82
- "Synchronizing Supporting Detail with Essbase" on page 83

Adding Supporting Detail

Use the Supporting Detail window to set and change how detail items aggregate to cell values in a data form. For information about using supporting detail, see "Working With Supporting Detail" on page 79.

- To add supporting detail that calculates values in a data form:
- 1 Open a data form in Excel, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 In the data form, select the data cells for which you want to add detail.

You can select a range of contiguous cells in a row or column in online mode only. You cannot select a section of cells that include a combination of rows and columns. To add detail in offline mode, you can select cells only on a cell by cell basis.

3 Select Hyperion, then Supporting Detail. For Office 2007, click Supporting Detail in the Task section of the Hyperion ribbon.

The Supporting Detail window is displayed. This window reflects your cell selection in the data form.

4 Replace the untitled text with an explanation or description.

You can include up to 1,500 characters of supporting detail. The text and its associated operator must be unique among children of the same parent.

5 Use the buttons to create or change the indented hierarchy to reflect the structure and calculations that you want.

For example, you can click Add Child to add a line item directly below the selected item.

For more information about the structure that you can create, see "Working with Supporting Detail Hierarchy" on page 81.

6 Set the mathematical relationships among the line items by selecting an operator for each line item.

You can select from these operators: + (add), - (subtract), * (multiply), / (divide), and \sim (ignore). Selecting Ignore saves the item text without requiring a numeric value.

7 Enter data for the items for which you want to set or calculate values.

When you enter numbers into the Supporting Detail window, use the scaling that was set up for the data values in the data form.

8 Click OK.

Values are dynamically calculated and aggregated before the data is saved.

Totaling When Supporting Detail Cells Are Blank

If a data cell in supporting detail is blank, the cell is not aggregated. Do not assume that a blank cell means zero.

For example, you might define the following supporting detail to calculate the daily rate of hiring an instructor (\$250) times the number of days per month for which you plan to retain an

instructor (4 in January, but none in February). As shown in the following example, the Instructor total for Feb is 250, even though you do not intend to hire an instructor in February:

| | Jan | Feb |
|------------|------|-----|
| Instructor | 1000 | 250 |
| Rate * | 250 | 250 |
| Days | 4 | |

To correctly total values in this situation, do not leave the cell blank. Instead, leave the Rate cell blank, or enter a zero in the relevant Days data cell, as shown below:

| | Jan | Feb |
|------------|------|-----|
| Instructor | 1000 | 0 |
| Rate * | 250 | 250 |
| Days | 4 | 0 |

In the solution above, the rate (250) is multiplied by 0 (zero), resulting in a value of zero.

Working with Supporting Detail Hierarchy

The supporting detail hierarchy should reflect the information that supports the cell values and the mathematical operators that create the appropriate relationships.

- To create or change the supporting detail hierarchy:
- 1 Open a data form in Excel, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 In the data form, select the cells with supporting detail with which you want to work.
- 3 Select Hyperion, then Supporting Detail. For Office 2007, click Supporting Detail in the Task section of the Hyperion ribbon.

The Supporting Detail window is displayed. This window shows your cell selection in the data form.

4 Create or change the rows in the hierarchy that provide the detail for the data values by clicking an item and selecting an option:

| Option | Result |
|--------------|--|
| Expand All | Shows all levels of supporting detail |
| Collapse All | Collapses all levels of supporting detail |
| Add Child | Adds a new line item one level below the selected line item. You can add an unlimited number of children, but keep in mind its potential impact on performance |

| Option | Result |
|---------------|--|
| Add Sibling | Adds a line item at the same level as the selected cell. You can add an unlimited number of siblings, but keep in mind the potential impact on performance |
| Delete | Removes the selected line item |
| Delete All | Removes all supporting detail at once |
| Promote | Moves the selected line item to the next-higher level in the hierarchy |
| Demote | Moves the selected line item to the next-lower level in the hierarchy |
| Move Up | Places the selected line item before its preceding sibling |
| Move Down | Places the selected line item after its sibling successor |
| Duplicate Row | Adds a row below the selected item, duplicating the structure of the selected row (text, operator, and values) |
| Fill | For rows, copies the data from the current cell to the cells to the right of the current cell |
| Сору | Copies data from the current cell |
| Paste | Pastes data to a cell |

5 Click Close.

The save operation stores the detail text and values and stores the aggregate values in the Essbase database.

Viewing or Changing Supporting Detail

Cells that have supporting detail are indicated in the Excel grid by a light green background.

You can select one cell or a range of contiguous cells in a row or column. You cannot select a section of cells that includes a combination of rows and columns. Select cells that are in the local currency if you need to write to them.

- To view or change supporting detail:
- 1 Open a data form in Excel, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 In the data form, select the data cells for which you want to view or change detail.
 - You can select one cell or a range of contiguous cells in a row or column. You cannot select a section of cells that includes a combination of rows and columns. Select cells that are in the local currency if you need to write to them.
- 3 Select Hyperion, then Supporting Detail. For Office 2007, click Supporting Detail in the Task section of the Hyperion ribbon.
- 4 View or change the line items or the calculations that aggregate the data in the selected cells.
 - For instructions about adding and maintaining supporting detail, see "Adding Supporting Detail" on page 80 and "Working with Supporting Detail Hierarchy" on page 81.

Synchronizing Supporting Detail with Essbase

In Planning applications, when you delete supporting detail for a cell, you affect the associated value in the relational database. You specify how to handle the stored Essbase value. You can set it to #Missing or leave it as it was before the supporting detail was deleted. This feature is useful if you want to use supporting detail as a scratch pad or calculator.

- To synchronize supporting detail with Essbase:
- 1 Open a data form in Excel, using the procedure described in "Opening Data Forms in Excel" on page 59.
- 2 In the data form, click the cell that has the supporting detail you want to remove.
- 3 Select Hyperion, then Supporting Detail. For Office 2007, click Supporting Detail in the Task section of the Hyperion ribbon.
- 4 In the **Supporting Detail** window, delete the information, then click **OK**.
- 5 Select an option from the displayed message to specify how to handle the aggregate value of the deleted supporting detail stored in Essbase:
 - To delete the value from Essbase, click Yes, set the value(s) to #Missing.
 - To leave the data value in Essbase as is, click No, leave the value(s) as is.

Working with Formulas in Data Forms

When working with data forms, you can create Excel formulas to manipulate or analyze the data. You can create formulas inside or outside the data form's grid. If you want to, you can save the formulas with the Excel workbook so that when you use the workbook again, the formulas are preserved.

Note:

Formulas that reference data within the same grid are not supported.

You can create formulas:

- Inside or outside the data form's grid
- On cells that have cell text

You cannot create formulas:

- On read-only cells or cells that are locked
- On cells that have supporting detail (in Planning) or line item detail (in Financial Management)

Formulas are preserved even when you:

- Refresh the data form without saving the data
- Later open the saved .XLS worksheet

• Expand or collapse rows or columns

Formulas interact with other functionality in the following ways:

- For Planning only: You can spread data values using a formula in a summary parent cell. Spreading logic ignores formulas in child cells. For example, you could enter a formula in Qtr1 whose calculated value is spread over Jan, Feb, and March. However, if you enter a formula in Feb, that cell is ignored when the value for Qtr1 is spread. For more information about how data is spread, see "Adjusting and Spreading Data Values" on page 72
- If you move a referential formula, its cell references are updated to reflect its new location
- If you use the Suppress #Missing Rows feature on a cell that has a referential formula, the cell references are not updated; the cell relationship is absolute, not relative
- If you use the Adjust feature to change the value of a data cell on which you have created a formula, the adjusted values overwrite the formula

If you create formulas in the data form, you are prompted to save the workbook as an .XLS file with the new formulas under the following conditions:

- You change the current page.
- You take a data form offline (this applies only to Planning).
- You select a different data form.
- You connect to a different data source.

Even if you save the formulas with the workbook, you lose temporary access to the formulas if:

- You change the current page, data form, or data source.
- You take the data form offline.

Working with Smart Lists

Data Sources: Planning, Essbase

A Smart List is displayed in grid or data form cell as an alphanumeric text description, but stored in the database as a number. For example, an integer Smart List for a reporting cycle may have values 1-5, for Yearly (1), Quarterly (2), Monthly (3), Daily (4), and Hourly (5). The values that are displayed on the data form are Yearly, Quarterly, Monthly, Daily, and Hourly. When you select one of these values on the form, and save the form, the number associated with it is stored in the database. This saves you from memorizing the numeric values associated with each time period in the reporting cycle.

You access Smart Lists from custom drop-down lists in cells. When you click into cells whose members are associated with a Smart List (as a member property), you select a drop-down list option instead of entering data. Smart Lists display in cells as down arrows that expand into lists when clicked.

For example, suppose a data form contains a Smart List named Justification that provides selections for Research, Customer Feedback, and Expansion. When you click into Account cells named Reason (whose members are associated with the Justification Smart List), a down arrow is displayed. When you click the down arrow, it expands into a drop-down list containing:

- Research
- Customer Feedback
- Expansion

Select one of the Smart List options as the value for the cell.

Note:

Smart Lists are defined and enabled by the Administrator.

- ➤ To select a Smart List:
- 1 In a grid or data from, click in a data cell containing a Smart List.

Note:

Only cells whose members are associated with Smart Lists (as a member property) contain Smart List drop-down lists.

A down arrow is displayed in the cell.

2 Click the down arrow and select an option. The option you select is entered into the data cell.

Query Designer

In This Chapter

| Query Designer Features | 87 |
|---|----|
| Query Designer Worksheets | 8 |
| Creating Queries from Default Reports | 88 |
| Extracting Queries from Reports | 89 |
| Editing Queries | 90 |
| Filtering Members in Queries | 90 |
| Filtering Data | 91 |
| Analyzing Time-Related Data in Query Designer | 92 |
| Rerunning Queries | 92 |
| MDX Queries | 93 |

Query Designer Features

Query Designer is used to design report layouts and select members for queries in Financial Management and Hyperion Enterprise data source types. For Essbase, Planning, and Oracle BI EE, see Chapter 5, "Smart Slices".

With Query Designer, you can:

- Design queries from default reports
- Extract queries from existing reports
- Enter and run MDX queries for advanced users
- Share queries with other users

Query Designer Worksheets

In Query Designer worksheets, some ad hoc operations are disabled when you design queries but are enabled in the reports.

Operations enabled only after running queries:

- Formulas
- Asymmetric reports
- Comments

87

- Blank rows or columns
- Changes to alias tables
- Ad Hoc actions such as zoom in and out, keep and remove only, and double-click

Operations allowed neither during query design nor after running the query:

- Filtering of column members
- Changing data sources in query mode

Note:

When using grids that contain duplicate (non-unique) members, the Member Name option must be selected in the Options dialog box to display the qualified name in the worksheet (for Essbase only). Ensure that duplicate members use fully-qualified member names.

You can create queries from a blank sheet. After retrieving a default report from an Essbase, Financial Management or Hyperion Enterprise data source, run Query Designer and modify the layout. See "Creating Queries from Default Reports" on page 88. Another use of Query Designer is taking an existing report, such as one containing the budget for West Coast markets, and extracting a basic query from the report. The query can then be modified. See "Extracting Queries from Reports" on page 89.

Options such as suppress missing rows, repeat member names, and indentation are defined as in ad hoc queries and can be set while working in the query sheet, but are not applied until after you run the query.

You cannot change data sources in the same query worksheet. However, you can have multiple Query Designer worksheets in the same workbook. For example, you can have one query sheet connected to an Essbase data source and another connected to a Financial Management data source.

Query Designer worksheets are designated by *Sheetname* - Query; for example, Sheet1 - Query. If you have more than one query sheet, the sheet is given a different name, such as Sheet2 - Query. After running a query, a report sheet is created with the name *Sheetname* - Report; for example, Sheet1 - Report, and the query sheet, *Sheetname* - Query, becomes hidden. Selecting Hyperion, then Query, and then Query Designer enables you to view and modify the query sheet.

Queries can be saved as .xls files. You can use queries as templates and share them with other users, who may want to run the same query or customize the query to their own needs.

Creating Queries from Default Reports

Starting with default reports, you can create queries.

- To create queries from default reports:
- 1 In Excel, connect to an Essbase, Oracle BI EE, or Financial Management or Hyperion Enterprise data source.
- 2 Click Query Designer or select Hyperion, then Query, and then Query Designer.

A worksheet called "Sheetname - Query" is created. It contains a default query (for example, Sheet1 - Query) and a POV toolbar that contains dimensions and a dropdown menu of dimension attributes.

3 From the POV toolbar, drag and drop dimensions and attributes either to columns or rows in the worksheet or within the POV itself. You can also enter members directly into the grid.

If you want to select members or filters for the dimensions, click the dimension name on the POV toolbar to invoke the Member Selection dialog box. See "Filtering Members in Queries" on page 90.

4 Click on the POV toolbar.

The resulting report is displayed in a new sheet called "*Sheetname* -Report;" for example, Sheet1 - Report. The report and query can be saved. The query is saved as part of the .xls file.

Note:

The .xls file that results from running the query can be saved for possible use as an Hyperion Enterprise or Essbase data load data source.

Extracting Queries from Reports

Use Query Designer to extract queries from existing reports, which you can modify and share with others.

- To extract queries from existing reports:
- 1 Open a report in Excel.
- 2 Connect to a Financial Management or Hyperion Enterprise data source.
- 3 Click Query Designer or select Hyperion, then Query, and then Query Designer.

A new sheet called "Sheetname - Query" is created. The sheet contains a query extracted from the existing report; for example, Sheet1 - Query.

4 Modify the query.

For example, you can delete members, add members, or pivot to create different queries and layouts.

Note:

Modifying the query does not preserve the format of the original report after running the query.

5 Click on the POV toolbar.

The resulting report is displayed in a new sheet called "*Sheetname* -Report"; for example, Sheet1 - Report.

Editing Queries

You can modify queries. After you run a query, the Query Designer sheet, *Sheetname* - Query, becomes hidden. You must run Query Designer again to view the Query Designer sheet.

To edit queries:

1 Select the Query Designer sheet Sheetname - Query; for example, Sheet 1- Query.

If the Query Designer sheet is hidden, select **Hyperion**, then **Query**, and then **Query Designer**. The Query Designer sheet is displayed.

- 2 Edit the query, such as adding new members, pivoting, removing members, and specifying data filters.
- 3 Click on the POV toolbar.

The report in Sheetname-Report is updated.

Note:

Rerunning queries regenerates the report, therefore any changes to the report, such as zooming in on members and entering comments or formulas are lost.

Filtering Members in Queries

You can filter members during member selection for queries for Essbase and Hyperion Enterprise. Filter members for Essbase by:

- Children
- Descendants
- Levels
- Generation
- UDA (user-defined attributes; defined by the administrator)
- Attribute
- Subset of attribute dimensions to create conditional expressions
- Period
- Date range

Filter members for Hyperion Enterprise:

- Account dimension members by:
 - Input
 - Calculated
 - o Dynamic
- Entity dimension members by:
 - o None

- Parent
- o Base
- Parent + Base + Subentity
- Parent + Subentity
- Base + Subentity
- Parent + Base
- Period dimension members by:
 - o None

 - Daily
 - o Weekly
 - Monthly
 - Quarterly
 - Trimester
 - Half-Yearly
 - Yearly

Using the Member Selection dialog box, specifying member filters inserts MDX syntax into the grid. For example, to filter by descendents for the East market, the MDX syntax is DESCENDANTS ([East]). You can modify the MDX statement by editing the cell. For example, you can change the statement to DESCENDANTS ([West]) to filter members by descendents of West. Other examples of MDX syntax include ([100].CHILDREN), ([Product].LEVELS (1).MEMBERS), and ATTRIBUTE([True]). You can change the values to ([200].CHILDREN), ([Product].LEVELS(2).MEMBERS), and ATTRIBUTE([False]). If you type an invalid MDX statement, an error message is displayed when you run the query. See "Filtering Members" on page 98.

Note:

MDX queries are not supported for Hyperion Enterprise data sources.

Filtering Data

You can filter data in queries for Essbase data sources only. Filtering data limits the amount of data returned to a specified top or bottom criterion. Top or bottom ranking enables you to view, for example, the top 10 products in sales for a given region.

- ➤ To filter data:
- 1 In the report, select a dimension.
- 2 Select Hyperion, then Query, then Data Filter.

- 3 From Data Filter, under Count, select Top or Bottom and specify a number.
- 4 Under Set, click and in Member Selection, select a row member for ranking, and click OK to return to Data Filter.
- 5 Under Value, click , and in Member Selection, select a column member to run the ranking against, and click OK to return to the Data Filter dialog box.
- 6 Click OK.

An MDX query, for example TopCount({ [Qtr3] }, 10, [Measures].[Profit]), that represents your data filtering settings is inserted into the grid. The example returns the top 10 most profitable products in quarter 3.

7 Click on the POV toolbar.

The query results are displayed in the sheet.

Analyzing Time-Related Data in Query Designer

(Essbase only) Using Smart View, you can analyze flash metrics such as sales of cost of goods sold against time-based metrics. This enables you to look for trends, find averages for different time periods, and so forth. To do this, you use linked attributes which enable periodicity of members. Periodicity is a shared pattern among time dimension members that make them meaningful for time-based analysis (January and April share periodicity as first months of quarters, for example). Day by month, day by week, and week by year are examples of linked attributes. You can also set ranges for linked attributes and apply filters.

- To analyze time-related data in Query Designer:
- 1 Create a query as described in "Creating Queries from Default Reports" on page 88.
- 2 From the Query Designer toolbar, select Date-Time dimension and drag it to the grid or within the toolbar.
- 3 Click **Date-Time** in the Query Designer toolbar to open **Member Selection**, where you can select members and apply Period, Range, and other filters as described in "Filtering Members in Queries" on page 90.
- 4 Under Attributes on the Query Designer toolbar, select an attribute or linked attribute in the dropdown menu, then drag it to the grid or within the toolbar. Repeat as necessary for other attributes.
- To select members and apply filters to an attribute, click the attribute name on the Query Designer toolbar to open Member Selection.
- 6 Click on the POV toolbar.

Rerunning Queries

You can use **Hyperion**, then **Refresh** or **Refresh** All to update data on query reports, but refreshing does not rerun queries. For example, if data filtering is set to return the top 10 cars sold in California, the sales data for the 10 cars is refreshed, but the set of 10 cars does not change.

To update the top 10 list of cars, you must rerun the query. Rerunning queries regenerates reports, therefore changes to reports, such as zooming in and entering comments, are lost.

MDX Queries

Advanced users can bypass the Query Designer interface and enter MDX commands in the query sheet or in the Execute MDX dialog box.

- ➤ To execute MDX queries:
- 1 In Excel, connect to an Essbase data source.
- 2 Select Hyperion, then Query, then Execute MDX.
- 3 In Execute Free Form MDX Query, enter the MDX query.

For example:

```
SELECT {[Sales], [Cogs]} on columns, Filter ([Product].Levels(2).Members, AVG([Year].CHILDREN, 9001.0) > 9000.00) on rows
```

4 Click Execute.

9

Working with Ad Hoc Analysis

In This Chapter

| The Ad Hoc Grid | 96 |
|---|-----|
| Setting Ad Hoc Options. | 97 |
| Selecting Members for Ad Hoc Analysis | 97 |
| Working with Dynamic Time Series Members | 102 |
| Formatting Data Cells in the Grid | 103 |
| Refreshing the Grid | 104 |
| Zooming In on Dimension Members | 104 |
| Zooming Out on Dimension Members | 105 |
| Retrieving Attribute Dimensions and Members | 105 |
| Viewing the Qualified Name of a Member | 106 |
| Displaying Aliases for Member Names | 107 |
| Accessing Data from a Hybrid Analysis Relational Source | 108 |
| Pivoting Dimensions | 108 |
| Keeping Data | 109 |
| Removing Data | 110 |
| Adding Cell Text in Ad Hoc Grids | 110 |
| Viewing Cell Text in Ad Hoc Grids | 111 |
| Calculating Data | 111 |
| Selecting a Calculation Script | 112 |
| Inserting Rows and Columns | 112 |
| Translating Data | 113 |
| Consolidating Data | 113 |
| Submitting Data | 114 |
| Working with Formulas in Ad Hoc Grids | 114 |
| Drill-Through Reports | 115 |
| Data Perspective | 115 |
| Member Perspective | |

Ad hoc grids are used to analyze data from a data source. An ad hoc query is a request to the database server to search its data for specific information or results. Ad hoc queries enable you to view data for specific dimensions from the data source without writing spreadsheet functions.

Ad hoc analysis is supported for Essbase, Hyperion Enterprise, Oracle BI EE, and Financial Management data source providers.

The Ad Hoc Grid

Note:

Smart View recommends a maximum grid size of 65,000 cells.

After you connect to a data source and open a worksheet, you can open the default Ad Hoc grid by double-clicking a cell in the worksheet or by selecting Refresh to create a default grid. To open a grid by double-clicking, you must have enabled this feature from the Options dialog box. For more information on ad hoc options, see "Ad Hoc Options" on page 207.

The default grid layout is specific to the data source. You can begin working with data using the default grid, or you can update the columns and rows with additional or different dimension members. For more information, see "Selecting Members for Ad Hoc Analysis" on page 97.

If you connect to a different data source while you are working in a grid, the system displays a message that you have applied a new connection to a grid that was connected with a different connection. You can continue to work with the new connection or either disconnect and then reconnect to the proper data source or select the data source from the Active Connections menu option. If you are working in a new grid, to reapply the original data source, you must reconnect and then refresh the grid. If you are working in a grid from a saved workbook, to reapply the original data source, you can reconnect to the saved data source without refreshing the grid.

Note:

If you are working in a new grid and you change the data source connection, you must refresh the grid in order to view the POV. If you open a spreadsheet with a previously saved POV, change the data source, and refresh the sheet, the POV is not updated with the new data source members. You must change the POV when you change the data source.

You can change the POV settings at any time for a grid. If you want to save the POV settings in a worksheet, after you make any changes to the POV, you must refresh the settings before you save the worksheet. For more information, see "Refreshing the Grid" on page 104.

Note:

If you make changes to the POV of a saved workbook after changing the data source connection, the changed POV will not be saved when the workbook is saved.

You can also change the starting dimensions in the default grid by selecting a different dimension or member in the POV Manager before performing a Refresh, as described in "Editing the Point of View" on page 221.

You can dynamically change the members defined on the rows or columns of the grid. You can select a member on a row or column and use the Member Selection option to replace the selected member. For more information see "Selecting Members for Ad Hoc Analysis" on page 97.

Note:

When using ad hoc grids with Financial Management, invalid cells that contain a value display the value as zero. This is different from other Financial Management modules where invalid cells that contain a value display that value.

Setting Ad Hoc Options

You can set options to control how data is retrieved into the ad hoc spreadsheet and which data is retrieved as described in "Ad Hoc Options" on page 207.

Selecting Members for Ad Hoc Analysis

You can select the dimensions and members for the rows and columns of the ad hoc grid. After you select members, the grid is updated with the new dimension members.

You use the Member Selection dialog box to select dimension members. You can filter the list of members and you can view labels or descriptions for dimension members. For example, when you select to view descriptions, the P_Series member is displayed as "P_Series - Phones and PDAs".

You can select a maximum of 1,000 members for the ad hoc POV.

You can also search for dimension members. For more information, see step 11 on page 98.

- To select dimension members:
- 1 Select Hyperion, then Member Selection.

The Dimension Name Resolution dialog box is displayed.

2 From the **Dimensions** drop-down list, select a dimension and click **OK**.

The Member Selection dialog box is displayed.

- 3 Optional: To filter the list of members, select a member list from the Filter drop-down list. See "Filtering Members" on page 98.
- 4 Optional: For advanced filtering, highlight a member without selecting its check box, then select a member list.

This uses the highlighted member as the filter parameter. For example, if you highlight United States and change the filter list to Children, the system displays the children of United States.

- 5 Optional: To display dimension descriptions instead of labels, select Use Descriptions.
- 6 Optional: To display Dynamic Time Series members for a time dimension, select Dynamic Time Series Members.

Note:

The Dynamic Time Series option is available only when working with Essbase data sources. For more information on selecting and working with Dynamic Time Series members, see "Working with Dynamic Time Series Members" on page 102.

7 Optional: To display active entities only, select Active Members.

Note:

The Active Members option is available only if the application has been set up for Organization by Period. For information on Org by Period, see the Hyperion Financial Management – System 9 Administrator's Guide.

- 8 In the Members list, highlight a member, then use one of the following methods to select members:
 - To select individual members, select the check box next to each member that you want to use.
 - To select all members for the highlighted member, click
 - To select all immediate children for the highlighted member, click and from the drop-down list, select Children.
 - To select all base members for the highlighted member, click and from the drop-down list, select Base Members.
- 9 Click , to move the selected members to the **Selection** list.

Tip:

To remove members from the Selection list, select the members, then click —. To remove all members from the Selection list, click

- 10 Optional: To reorder members in the Selection list, click the Up or Down arrows or the Make Lowermost and Make Topmost buttons above the list to move members.
- 11 **Optional:** To search for members in the selected dimension, use the Search button.
- 12 When you finish selecting members, click **OK**.

Filtering Members

When selecting members from a large outline, use filters to limit the members returned in a report. For Essbase, you can select attributes and set conditions for them using the Subset dialog box.

➤ To filter members:

- After selecting a member in the spreadsheet, access the Member Selection dialog box using several methods:
 - Select Hyperion, then Member Selection. For Office 2007, click Member Selection in the Task section of the Hyperion ribbon.
 - From POV toolbar, select the down arrow button next to a dimension and select the ellipses (...).
 - From the POV toolbar in Query Designer, double-click the dimension name or select the down arrow button next to a dimension and select the ellipses (...).
- 2 From **Dimension**, select a dimension.
- 3 From Filter, select an option:

Note:

Filter options vary depending on whether you are connected to Essbase, Financial Management, or Hyperion Enterprise data sources. Filter options for Essbase are static and are described in this procedure. Filter options for Financial Management are dynamic.

- None—Applies no filters
- Children—Selects only children of a specific member
- Descendants—Selects only descendants of a specific member
- Level—Selects members by level number
- Generation—Selects members by generation number
- UDA—Selects members by user defined attributes (Essbase only)
- Attribute—Selects members associated with a specified attribute dimension (Essbase only)
- Subset—Opens the Subset dialog box for specifying attribute dimension criteria (Essbase only)
- Input—Selects input accounts (Hyperion Enterprise only)
- Calculated—Selects calculated accounts (Hyperion Enterprise only)
- Dynamic View—Selects dynamic view accounts (Hyperion Enterprise only)
- Parent—Selects parent entities (Hyperion Enterprise only)
- Base—Selects base entities (Hyperion Enterprise only)
- Subentity—Selects subentities (Hyperion Enterprise only)
- Daily—Selects daily entries (Hyperion Enterprise only)
- Weekly— Selects weekly entries (Hyperion Enterprise only)
- Monthly—Selects monthly entries (Hyperion Enterprise only)
- Quarterly—Selects quarterly entries (Hyperion Enterprise only)
- Trimester—Selects trimester entries (Hyperion Enterprise only)

- Half-Yearly—Selects half-yearly entries (Hyperion Enterprise only)
- **Yearly**—Selects yearly entries (Hyperion Enterprise only)
- Varying Attribute—see "Member Perspective" on page 117

If you selected:

- None, Children, Descendants, Input, Calculated, Dynamic View, Parent, Base, Subentity, Daily, Weekly, Monthly, Quarterly, Trimester, Half-Yearly, or Yearly, go to step 9.
- Level or Generation, the Filter Arguments dialog box is displayed. Go to step 5.
- UDA, the Filter Arguments dialog box is displayed. Go to step 6.
- Attribute, the Filter Arguments dialog box is displayed. Go to step 7.
- Subset, the Subset dialog box is displayed. Go to step 8.

In Filter Arguments:

- Specify the level or generation number in Level or Generation.
- b. Click OK.
- Go to step 9. c.

In Filter Arguments:

- From the UDA dropdown menu, select the user-defined attribute to use.
- h. Click OK.
- c. Go to step 9.

7 In Filter Arguments, in Attribute Members select

The Subset dialog box is displayed, enabling you to specify an attribute so that members are returned containing the specified attribute.

- In Dimension, select an attribute dimension. a.
- h. In Member, select an attribute member.
- Click Add. c.
- d. Optional: You can change the selected attribute dimension and member. Click Set after making the changes. Click Delete to remove the attribute.
- Go to step 9. e.

In Subset, define attribute conditions to return members meeting the condition.

- In Dimension, select an attribute dimension; for example, Caffeinated. a.
- b. In Member, select an attribute member; for example, True.
- Click Add. c.
- d. In **Dimension**, select another attribute dimension; for example, Pkg Type.
- In Member, select another attribute member; for example, Bottle. e.
- f. Click Add.

An AND condition statement is created; for example, [True] AND [Bottle].

- **Optional:** Select the condition statement to change it. g.
- h. Select **Operator** and a condition:
 - AND—The selection must meet both conditions; for example, the members must be caffeinated and packaged in bottles.
 - OR—The selection must meet either conditions; for example, the members can be caffeinated or packaged in bottles.
 - NOT—The selection includes only the members that do not have the selected condition; for example, the members that are not caffeinated.
- i. Optional: You can nest conditions by selecting more attributes and selecting Add, then to Root.
- Click OK after setting the conditions to return to Member Selection. j.
- In **Member Selection**, click to add the members to the selection.
- 10 Click OK.

Member Selection Differences

The Member Selection dialog box can be accessed in several ways. However, depending on where you accessed the Member Selection dialog box, the functionality can be different.

Table 5 Member Selection Functionality Differences

| Member Selection Type | Purpose |
|-----------------------|--|
| Ad Hoc | Accessed from Member Selection on the Hyperion menu. You can select a dimension on the grid and change which members are selected for the report. You cannot apply member filters. |
| POV toolbar | Accessed from POV toolbar. You can change the dimension or member to make it available on the POV toolbar, so that you can add it to the grid. |
| Query Designer | Accessed from Query Designer. You can change the members for the query. Member filtering is available. |

Dragging and Dropping Dimension Members

You can drag and drop dimensions between rows and columns and move them in and out of the grid. You can drag and drop the dimensions to any row or column on the grid.

Note:

If you drag a dimension over an existing dimension in the grid, the new dimension is placed on the outer side of the existing dimension.

- To drag and drop a dimension member, use one of the following methods:
 - To drag a dimension member from the Point of View toolbar to the grid, select the member, and drag it to the desired location on the grid.
 - To drag a dimension member from the grid to the Point of View toolbar, select the member in the grid, right-click and drag it to the toolbar.

Note:

Smart View drag and drop is disabled after you perform Excel editing actions such as typing over member values, inserting or deleting columns or rows, dragging and dropping an Excel region. etc. To re-enable Smart View drag and drop, refresh the grid.

Working with Dynamic Time Series Members

Note:

This option is available only when working with Essbase data sources.

Dynamic Time Series members are predefined members used in to-date calculations. Dynamic Time Series members do not appear as members in your database outline; instead, they represent a generation in a Time dimension. For example, in the Essbase Sample Basic database, the Application Designer can create a generation name called Quarter for generation 2 of the Year dimension that contains the data for Qtr1, Qtr2, Qtr3, and Qtr4. When you create the generation name Quarter, Essbase creates and enables a Dynamic Time Series member called Q-T-D.

To use Dynamic Time Series in calculations, you first define the latest time period for which you want data. The latest time period is the level 0 member in a Time dimension. For example, in the Sample Basic database, the level 0 members are the months of the year: Jan, Feb, Mar, and so on. If the current month is August, and you want to know the sales data for the quarter up to the current month, Dynamic Time Series calculation gives you the sales data for the months of July and August.

Note:

If you are manually entering Dynamic Time Series members into a free-form grid, formats such as Q-T-D(Jan), QTD(Jan), YTD(Mar), Y-T-D(Mar), MTD(Jun), and M-T-D(Jun) are supported. Formats such as QTDate(Jan) are not supported.

- To specify the latest time period to use:
- 1 Access the **Member Selection** dialog box using one of the following methods:
 - Select Hyperion, then Member Selection. For Office 2007, click Member Selection in the Task section of the Hyperion ribbon.

From the Options drop-down menu on the POV toolbar, choose Select Members, select the time dimension (if available), and then choose a member.

The Member Selection dialog box is displayed, and the Time dimension and member names are displayed in the Members list box.

2 Select the **Dynamic Time Series Member** check box.

The dimension and member names are displayed as Dynamic Time Series members.

The following are valid Dynamic Time Series members:

| Dynamic Time Series | Generation Name |
|---------------------|-----------------|
| H-T-D | History |
| Y-T-D | Year |
| S-T-D | Season |
| P-T-D | Period |
| Q-T-D | Quarter |
| M-T-D | Month |
| W-T-D | Week |
| D-T-D | Day |

- 3 Select the Dynamic Time Series members that you require from the **Members** list box and click
- In the **Selection** list box, select a Dynamic Time Series member to highlight it, and then click the **dts** button.
- 5 From the drop-down list in the Select DTS Member dialog box, select the latest period on which the to-date calculation will be based.
- 6 Click OK.
- Repeat step 4 through step 6 for all Dynamic Time Series you have added to the Selection list box.
- Click OK to close the Member Selection dialog box.

Formatting Data Cells in the Grid

In both free form grids and structured grids, you can use the Capture Formatting command to capture and apply formatting styles, such as font color or number styles, to selected data cells.

If you are working in a free form grid, you must refresh the grid after you type member names in the grid in order to use Capture Formatting.

Note:

The procedure in this topic assumes you are displaying the Formatting toolbar in Excel. To display the formatting toolbar, right-click in the toolbar area of Excel and select Formatting.

- To apply formatting to one or more data cells in a grid:
- 1 Select the cell or cells to which you want to apply formatting.
- Use Excel formatting options to format the cell or cells.
- Select Hyperion, then Capture Formatting to retain the formatting options for the selected cells.

Refreshing the Grid

You can refresh the grid at any time to retrieve data from the connected data source into the active worksheet. For example, after you change the Point of View, you can refresh the grid to see the changes. If you want to save the Point of View settings in a worksheet, you must refresh the settings before you save the worksheet.

Note:

To refresh a grid, 250 bytes per cell of memory is needed. For example, to refresh a 100x30 cell grid requires 750,000 bytes of JVM memory.

- To refresh the grid, take one of the following actions:
 - Select Hyperion, then Refresh. For Office 2007, click Refresh in the Review section of the Hyperion ribbon.
 - Click the Refresh toolbar button.
 - From the Point of View dialog box, click the Refresh arrow.
 - Right-click, then select Hyperion, then Refresh.

To refresh all grids, functions, and forms in the workbook, select Hyperion, then Refresh All. For Office 2007, click Refresh All in the Review section of the Hyperion ribbon.

Zooming In on Dimension Members

You can drill down to various levels of data in the grid by expanding the members. For example, if you want to view data for a specific quarter or month rather than for the whole year, you can zoom in on the [Year] member to see more detailed data.

When you zoom in on a member, it is expanded according to the options specified in the Options dialog box. If you select the option to zoom in on all levels, the grid displays all descendants of the selected member.

The default zoom option is to zoom in to the next level to retrieve data for the children of the selected member or members. For example, if you double-click the [Year] member, the system displays Quarter1, Quarter2, Quarter3, and Quarter4.

You can zoom in on the bottom level to retrieve data for the lowest level of members in a dimension. You can select retrieving data for the sibling level, the same level, or the same generation as the selected member.

Hybrid Analysis gives the spreadsheet user the ability to drill down from multidimensional members to relationally-stored members. In Smart View, when you connect an Essbase data source, you can drill down to relationally-stored members if they are present in the Essbase database. Drilling down on Hybrid Analysis members is enabled by default. For more information on Hybrid Analysis, see "Accessing Data from a Hybrid Analysis Relational Source" on page 108.

- To zoom in on dimension members, take one of the following actions:
 - Double-click on a cell.
 - Select a worksheet cell for which you want to zoom in, then select Hyperion, then Ad Hoc Analysis, then Zoom In. For Office 2007, click Zoom In in the Ad Hoc Analysis section of the **Hyperion** ribbon.

Zooming Out on Dimension Members

You can drill up to higher levels of data in the grid by collapsing, or zooming out to, dimensions. For example, if you previously drilled down on a dimension such as Period to view data for October, November, and December in Quarter 2, you might need to drill up to view aggregate data for the period. When you zoom out from December, the Period dimension is collapsed for October, November, and December and replaced with Quarter 2.

Note:

Zooming out is not supported for Oracle BI EE.

- To zoom out on dimension members:
- 1 Select a worksheet cell for which you want to zoom out.
- 2 Select Hyperion, then Ad Hoc Analysis, then Zoom Out. For Office 2007, click Zoom Out in the Ad Hoc Analysis section of the Hyperion ribbon.

Tip:

To zoom to the top level, double right-click on a dimension member.

Retrieving Attribute Dimensions and Members

You retrieve attribute dimensions or members by first performing an ad hoc query, then typing their names directly on the worksheet. The base dimension must already exist on the worksheet before you type the attribute name in the sheet.

For information on retrieving attributes in free-form grids, see:

- "Working with Attribute Dimensions" on page 122
- "Valid Grid with POV Region and Attribute" on page 134

- "Retrieving Attribute Dimensions in Free-Form Mode" on page 137
- To retrieve an attribute member by typing its name:
- 1 Perform an ad hoc query to place the base member on the worksheet.
- 2 Type the attribute name over the base dimension name in the worksheet.

The attribute member name replaces the base dimension name.

- Refresh the worksheet.
- 4 Zoom in on the attribute member name.
- 5 Optional: Perform other ad hoc query operations, such as pivoting or adding other dimensions to the sheet.

For example, to retrieve the Pkg Type attribute from the Essbase Sample Basic application and database, perform the following steps:

- Connect to Sample Basic in a blank worksheet and refresh the sheet. a.
- b. Drag the Product dimension from the POV and drop it on the Year dimension. Or, in the POV, select Options, then Pivot to Row, and then Product.
- In the worksheet, in the cell that contains the Product dimension, type Pkg Type over the c. word Product.
- d. Double-click Pkg Type to view measures by can or bottle package type.

Viewing the Qualified Name of a Member

As you perform ad hoc queries in the grid, you can view the *qualified* name for a member.

When you view the qualified name of the member, you are able to view the member name and the names of its parents up to the level in the dimension that uniquely defines the member. This can be helpful when querying on a grid that contains aliases or duplicate member names.

The qualified member name is displayed in a pop-up dialog box in the following format:

```
[Dimension].[Member]
```

The number of members displayed in the qualified name depends on the number of member levels necessary to provide the qualified name of the member.

- To view the qualified name of a member in the grid:
- 1 Select a member in the grid.
- 2 Select Hyperion, then Ad Hoc Analysis, then View Qualified Member Name.

A dialog box containing the qualified member name is displayed.

- 3 Click OK.
- 4 Optional: To view the qualified name of another member, repeat the procedure.

Note:

If you want to view qualified member names in the worksheet, select the "Name Only" option in the Display tab of the Options dialog box. See "Scale and Decimal Display Options" on page 215 for instructions on setting this option.

Displaying Aliases for Member Names

Note:

This option is not available when working with Hyperion Enterprise data sources.

Aliases are alternate names for database members. You can perform ad hoc retrievals using the database member name, which is often a stock number or a product code, or an alias, which can be more descriptive. Each database to which you connect can contain up to 10 alias tables.

For example, members of the Product dimension in the Essbase Sample Basic database are defined as codes, such as 100 and 200. A descriptive value for each member of Product, such as Colas and Root Beer, is defined in an alias table. In some cases, alias names may vary depending on the combination of other database members. For example, a Product member may have an alias in multiple languages for worldwide users.

Note:

Using the Change Alias Table command may result in the loss of cell comments and formulas that you added to the worksheet, and formatting that you preserved using the Capture Formatting command.

- To display aliases for member names:
- 1 In the Options dialog box Display tab, verify that in the Member Name Display Options group, either Member Name and Description or Description Only has been selected.

The "Member Name and Description" option or the "Description Only" option must be selected in order for alias names to display in the ad hoc grid. See "Display Options" on page 212 for information on setting the Member Name Display Options.

- 2 Perform one of the following options:
 - Select Hyperion, then Data Source Manager. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.

If you are using Essbase, you can select or change an alias table for a data provider connection in the Data Source Manager. New data sources and new worksheets associated with this connection use the specified alias table. New worksheets that contain a previously existing grid, however, continue to use the previous alias table. Copy/pasted data points are not affected by changes to the alias table. When an alias tables is modified in the Data Source Manager, the background POV is updated with the new aliases.

Changing the alias table in the Data Source Manager causes functions that are dependent on the connection to fail.

- Select Hyperion, then Ad Hoc Analysis, then Change Alias Tables. Use this option to change alias tables for existing grids.
- 3 In the Data Source Manager, right-click the active provider and select Change Alias Table.
- 4 In the Select Alias Table dialog box, select the alias table that contains the member names that you want to use, and click OK.
- Refresh the grid to display the alias names from the table you have chosen.

Use any of the methods described in "Refreshing the Grid" on page 104.

Accessing Data from a Hybrid Analysis Relational Source

Because relational databases can store several terabytes of data, they offer nearly unlimited scalability. Essbase multidimensional databases are generally smaller than relational databases but offer sophisticated analytical capabilities. With Hybrid Analysis, you can integrate a relational database with an Essbase database and thereby leverage the scalability of the relational database with the conceptual power of the multidimensional database.

Hybrid Analysis eliminates the need to load and store lower-level members and their data within the Essbase database. This feature gives Essbase the ability to operate with almost no limitation on outline sizes and provides for rapid transfer of data between Essbase databases and relational databases.

In Smart View, you do not need to set a specific option order to retrieve data that is stored in the Hybrid Analysis relational source. This functionality is enabled automatically. Perform zoom in operations as described in "Zooming In on Dimension Members" on page 104.

Pivoting Dimensions

Pivoting changes the orientation of the data on the worksheet. You can move dimensions between rows and columns and move dimensions in and out of the grid. When you move between rows and columns, the system moves the selected dimension to the outermost row or column on the opposite axis. For example, when you select Pivot to Row, the system moves the dimension to the far left side of the grid, and when you select Pivot to Column, the system moves the dimension to the top of the grid. You can also select to pivot a dimension out of the ad hoc grid to the Point of View.

Before you remove a dimension from a grid, you must first pivot the replacement dimension into the grid.

Note:

When the "Use Excel Formatting" option is selected (as described in "Using Excel Formatting" on page 215), member and numeric formats may unexpectedly change after pivot operations.

For example, member names may be centered and numeric values may be left justified. You can reset the grid to the proper format using the Excel formatting options.

- To pivot data, do one of the following tasks:
 - To move a dimension to the outermost row or column, click on a dimension, then select Hyperion, then Ad Hoc Analysis then Pivot. For Office 2007, click Pivot in the Ad Hoc Analysis section of the Hyperion ribbon.
 - To pivot a dimension to the Point of View, select the dimension, then from the Point of View toolbar, select Options, then Pivot to POV.
 - To pivot a dimension to a row, from the Point of View toolbar, select **Options**, then Pivot to Row, then select a dimension.
 - To pivot a dimension to a column, from the Point of View toolbar, select **Options**, then **Pivot to Column**, then select a dimension.

Note:

In Financial Management and Planning, you can pivot only from rows or columns with multiple dimensions. If you need to pivot a grid with one row dimension and one column dimension, first move a POV dimension to a row or column, pivot, then move POV dimension back.

Note:

In Hyperion Enterprise, the Pivot to POV, Pivot to Row, and Pivot to Column options are not available. in Oracle BI EE, only Pivot to POV is supported.

Keeping Data

When you select Keep Only, the system keeps only the selected member (the active cell) or member range within the grid. All unselected members are removed from the worksheet for that dimension.

The options that you select for member retention also affect the behavior of the Keep Only option. For more information, see "Member Retention Options" on page 211.

- To keep data in the grid:
- 1 Select the member or members that you want to keep.
- Select Hyperion, then Ad Hoc Analysis, then Keep Only. For Office 2007, click Keep Only in the Ad Hoc Analysis section of the Hyperion ribbon.

Removing Data

You can modify the ad hoc grid by removing dimension members from the grid. When you select Remove Only, the system removes the selected member (the active cell) and data or member range and data from the worksheet for that dimension. All unselected members are retained in the worksheet.

The options that you select for member retention also affect the behavior of the Keep Only option. For more information, see "Member Retention Options" on page 211.

Note:

If you delete members from a grid using the Delete key (not using the Remove Only command), you must also delete the corresponding data points; otherwise, when you refresh the grid, the remaining cells are treated as comments.

- To remove data from the grid:
- 1 Select the member or members that you want to remove.
- 2 Select Hyperion, then Ad Hoc Analysis, then Remove Only. For Office 2007, click Remove Only in the Ad Hoc Analysis section of the Hyperion ribbon.

Adding Cell Text in Ad Hoc Grids

Note:

This option is not available when working with Hyperion Enterprise data sources.

You can add a text description for any cell. For example, you might want to add an explanation of the cell's data. After you have added cell text, you can also make any necessary changes to the text.

- To add cell text:
- 1 Select the cell or cells for which you want to add text.
- 2 Select Hyperion, then Cell Text. For Office 2007, click Cell Text in the Task section of the Hyperion ribbon.

The Point of View is displayed for the cell that you selected.

3 In the **Cell Text** window, enter the text that you want to add.

You can add up to 2,000 characters of cell text for each cell. If you selected a range of cells, you can enter cell text in a separate text box for each cell.

4 Click OK.

Cells that contain text are indicated by dark blue borders.

Viewing Cell Text in Ad Hoc Grids

Note:

This option is not available when working with Hyperion Enterprise data sources.

If a worksheet cell contains cell text, you can view it from an ad hoc grid. Cells that contain text are indicated by dark blue borders.

- To view cell text:
- 1 Select the cell for which you want to view cell text.
- 2 Select Hyperion, then Ad Hoc Analysis, then Cell Text. For Office 2007, click Cell Text in the Task section of the Hyperion ribbon.

The Point of View is displayed for the cell that you selected.

3 View the cell text, then click **OK** to close the dialog box.

Calculating Data

When you are using an ad hoc grid, you can select several options to calculate data. You can calculate data, force a calculation to run for selected cells, calculate contribution values, or force calculation to run for selected contribution values.

To calculate data, you must have security access rights to the data. If the calculation option is not available for the cell, the system ignores that cell.

All options described in this topic are available to Financial Management users. For more information on calculating data from a Financial Management data source, see the Hyperion Financial Management – System 9 User's Guide.

Note:

If you are connected to an Essbase data source, only the Calculate option is available. Choosing this option invokes the Calculation Scripts dialog box, where you may select the calculation script that you want to run. See "Selecting a Calculation Script" on page 112 for more information.

Note:

If you are connected to Hyperion Enterprise data source, only the Calculate option is available. For Hyperion Enterprise data sources, refresh the data forms after calculating data to update the calculated accounts.

- To calculate data:
- 1 Select a cell or range of cells for which you want to calculate data.
- 2 Select Hyperion, then Calculation Options, then select one of the following options:
 - To calculate the selected cells, select Calculate.

Note:

If you are connected to an Essbase data source, you are prompted to select a calculation script to run, as described in "Selecting a Calculation Script" on page 112.

- To force calculation to run for all selected cells, select Force Calculate.
- To calculate contribution values, select Calculate Contribution.
- To force calculation to run for all selected contribution values, select Force Calculate Contribution.

Selecting a Calculation Script

If you are connected to an Essbase data source, you can choose a calculation script to calculate an Essbase database.

- To calculate the database with a calculation script:
- 1 Select Hyperion, then Calculation Options, then Calculate to display the Calculation Scripts dialog box.

The Cube area displays the databases that belong to this application. The calculation scripts associated with each cube are listed in the lower portion of the dialog box.

Tip:

To view a subset of scripts, clear all check boxes in the Cube area of the dialog box and then select the cube with which you are working to list only the scripts available for that cube in the lower portion of the dialog box.

2 Click on a calculation script name and click Launch.

A status message is displayed telling you whether the calculation was successful or not. If the calculation was not successful, contact your Essbase administrator.

Inserting Rows and Columns

In ad hoc grids, calculating and non-calculating columns and rows may be inserted within or outside the grid, Inserted rows and columns may contain formulas, text, or Excel comments, and are retained when you refresh or zoom in.

Translating Data

Note:

This option is not available when working with Hyperion Enterprise data sources.

In ad hoc grids, you can convert values from one currency to another by translating data. To do this, you must have security access rights to the data. For more information on translating data, see the *Oracle Hyperion Financial Management, Fusion Edition User's Guide*.

- To translate data:
- 1 Select a cell or range of cells.
- 2 Select Hyperion, then Calculation Options, then select an option:
 - To translate the selected cells, select **Translate**.
 - To force translation to run for all selected cells, select Force Translate.

Consolidating Data

When you are using an ad hoc grid, you can consolidate data for cells. You can consolidate selected data, consolidate all data, or consolidate all cells that contain data.

To consolidate data, you must have security access rights to the data and you must be assigned the Consolidate security role. To Consolidate All data, you must be assigned the Consolidate All security role.

If the consolidation option is not available for the cell, the system ignores that cell.

For more information on consolidating Financial Management data, see the *Hyperion Financial Management – System 9 User's Guide*.

- To consolidate data:
- 1 Select a cell or range of cells for which you want to run consolidation.
- 2 Select Hyperion, then Calculation Options, then select one of the following options:
 - To consolidate data for the selected cells, select Consolidate.
 - To consolidate data for all cells, regardless of whether they contain data, select Consolidate All.
 - To consolidate data for all cells that contain data, select Consolidate All With Data.

Submitting Data

After you calculate, translate, or consolidate data on an ad hoc grid, you can submit data from the grid to the data source to ensure that the data source uses the latest information from the worksheet. See "Submitting Data" on page 68.

To submit data to the data source, select Hyperion then Submit Data. For Office 2007, click **Submit** in the **Review** section of the **Hyperion** ribbon.

Note:

When the grid layout is changed (such as by inserting a column or row) or when the area outside of the data region is changed, the Submit Data option will be disabled. Select Hyperion then Refresh to refresh the grid so that the data region can be established by Smart View, then the Submit Data option will be enabled and data can be changed and submitted.

Working with Formulas in Ad Hoc Grids

You can use Excel formulas with ad hoc grids as follows:

- Inside or outside the ad hoc grid
- On cells that contain cell text or are locked

You cannot create formulas:

- On read-only cells
- On cells that have supporting detail (applies only toPlanning data forms)

Formulas are preserved even when you:

- Refresh the ad hoc grid
- Zoom in or zoom out (referential formulas are updated with their new relative positions)

Note:

After performing a Zoom In or Zoom Out operation, the location of the formula is calculated based on the first occurrence of a member on any row or column. If member names are repeated, then zooming in or zooming out on the grid may cause the formula location to be wrongly calculated. Oracle recommends that you avoid use of Zoom In and Zoom Out operations when working with grids containing repeated members and formulas referring to them.

- Use the Keep Only or Remove Only functions
- Add comments

Formulas are not preserved if you change the grid by:

- Pivoting data (for example, by selecting Pivot to POV or Pivot to Column)
- Clearing the grid
- Retrieving data without saving the formulas

Using the Change Alias Table command

Drill-Through Reports

Data Source: Essbase

Note:

Drill-though functionality is available only for Essbase cubes that were created in Oracle Essbase Studio or Oracle Essbase Integration Services.

The data stored in Essbase is summarized and calculated from detailed data stored in a relational database. For applications created in Essbase Studio, drill-through reports, which are predefined by administrators in Oracle Essbase Studio, enable you to drill through the Essbase data to view the detailed data in the relational database. For example, if you are analyzing retail sales for the first quarter in the Eastern region (Essbase data) you might want to see a list of customers who have purchase a particular product in a particular size (detailed data). A drill-through report could be created for this purpose.

Drill-through reports are associated with individual member cells and data cells; cells can be associated with multiple drill-through reports. Member and data cells that are associated with drill-through reports are indicated by a colored cell background and by tooltips that display a list of available drill-through reports when you mouse over a cell.

The data displayed in a drill-through report is dynamic.

- To access a drill-through report:
- 1 Select a member or data cell associated with a drill-through report.
- 2 Select Hyperion, then Ad Hoc Analysis, then Drill-through Reports to display the list of reports associated with the cell.
- 3 Select a report and click Launch. The detailed data associated with the member or data cell is displayed in the grid.

Data Perspective

Note:

Data perspective may not be enabled in your Smart View system. Your options for data perspective are enabled and configured by the Administrator in Administration Services.

Data perspective enables you to specify the perspective to use for viewing data of varying attributes, which are dimension attributes that vary with respect to independent continuous and discrete dimensions. For example, suppose a cola product is sold in both cans and bottles in

several different geographical markets over the course of a year. If the packaging (cans or bottles) varies depending on the market or changes from one type to the other during the year, the packaging type is a varying attribute. The data associated with the cola would be different depending on the time of year and the market

- To specify data perspective:
- 1 Select Hyperion, then Data Perspective.
- 2 From Perspective, under Selection, select an option (see Data Perspective Illustration for examples of options):
 - **Reality** to display the data with no perspective.
 - Last to display the data for the last level 0 member of each continuous independent dimension. For example, if Year is the continuous dimension and December is the last member of Year, then the data for December is displayed.
 - **Start** to display the data for the first level 0 member of each continuous independent dimension. For example, if Year is the continuous dimension and January is the first member of Year, then the data for January is displayed.
 - Custom if you want to specify both continuous and discrete members. For this option, select a Varying Attribute from the dropdown list. Then, for the dimensions listed under Independent Dimension, select members under Members. If you select Set Dimensions Only, all independent dimensions across all varying attribute are displayed, enabling you to apply a common perspective to all.
- 3 Click **OK**, then refresh the grid.

Data Perspective Illustration

In our example of cola sold in cans and bottles, suppose the Administrator has specified the following attributes for the cola packaging types to reflect how the cola was sold in Texas and California markets during the year:

- Can: California, January—December year
- Can: Texas, July—December
- Bottle: Texas, January—June

Figure 3 illustrates the Reality perspective. The data shown for California and Texas is data for the entire year. Since bottles were not sold in California, no data is returned (indicated here by "#Meaningless).

Figure 3 Data Perspective: Reality

| | A | В | С | D |
|---|----------|--------------|-------|--------|
| 1 | | California | Texas | Market |
| 2 | Bottle | #Meaningless | 405 | 405 |
| 3 | Can | 1587 | 234 | 1821 |
| 4 | Pkg Type | 1587 | 639 | 2226 |

Figure 4 illustrates the Last perspective and displays data for cans for California and Texas, but none for bottles, because bottles were sold only January through June in Texas.

Figure 4 Data Perspective: Last

| | А | В | С | D |
|---|----------|--------------|--------------|--------------|
| 1 | | California | Texas | Market |
| 2 | Bottle | #Meaningless | #Meaningless | #Meaningless |
| 3 | Can | 1587 | 234 | 1821 |
| 4 | Pkg Type | 1587 | 234 | 1821 |

Figure 5 illustrates the Start perspective and displays data for January. Bottles but not cans were sold in Texas in January, so only data for bottles is displayed. Cans but not bottles were sold in California in January, so only data for bottles is displayed.

Figure 5 **Data Perspective: Start**

| | A | В | C | D |
|---|----------|--------------|--------------|--------|
| 1 | | California | Texas | Market |
| 2 | Bottle | #Meaningless | 639 | 639 |
| 3 | Can | 1587 | #Meaningless | 1587 |
| 4 | Pkg Type | 1587 | 639 | 2226 |

Member Perspective

In addition to Data Perspective, you can specify member perspective for varying attributes. You do this when you are selecting members by using the Varying Attribute filter.

Note:

Member perspective may not be enabled in your Smart View system. Your options for member perspective are enabled and configured by the Administrator.

- To specify member perspective:
- 1 From Member Selection, under Filter, select Varying Attribute.
- 2 In Filter Arguments, click
- Specify an attribute to set the perspective and click **OK**.
- 4 In Varying Attribute Args under Varying Attribute, click the ellipsis button.
- 5 In **Subset**, in **Dimension**, enter an attribute dimension.
- In Member, enter an attribute member and click Set.
- Click OK.
- In Varying Attribute Args, under Perspective, click the ellipsis button.

9 From **Perspective**, select one of the following:

- Snapshot. One set of independent dimension members can be selected to identify the members of base dimension associated with the varying attribute. Here the start and end tuple are same.
- Range. A finite range of independent dimension members can be selected. A range can be specified only for continuous independent dimensions ("Year" would be an example). For discrete independent dimensions you can make only a single selection.
- 10 Click OK.

10

Using Free-Form Reporting to Retrieve Data

In This Chapter

| About Free-Form Reporting | 119 |
|--|-----|
| Free-Form Guidelines | 121 |
| Retrieving in Free-Form Mode. | 135 |
| Resolving Dimension Names. | 136 |
| Retrieving Attribute Dimensions in Free-Form Mode | 137 |
| Resolving Member Names in Free-Form Grids | 138 |
| Answering Business Questions Using Free-Form | 139 |
| Constructing an Example Free-form Report Using a Hyperion Enterprise Data Source | 145 |
| Analyzing Time-Related Data in Free-Form Mode | 150 |
| Highly Formatted Free-Form Report Example | 150 |
| Creating and Distributing Report Templates Using Free-Form | 151 |
| Retrieving Data into Asymmetric Reports | |

About Free-Form Reporting

In addition to Ad Hoc retrieval and Member Selection operations, Smart View supports freeform reporting.

Free-form reporting enables you to tell the connected data source specifically what you want to retrieve by typing member names into the worksheet. This free-form mode of reporting is especially useful when you are familiar with the dimensions and members in the database outline.

When you construct a report by entering names directly into a worksheet, Smart View queries the connected data source, interprets the member names entered, and creates a default view that is based on the location of the labels.

Free-form mode also provides most of the retrieval capabilities types of retrieval capabilities, enabling you to incorporate into the sheet many of the Ad Hoc retrieval techniques described in this chapter, such as zoom in and zoom out, pivot, and member selection.

With free-form, you can:

- Type dimension or member names directly into the grid
- Expand what you type using the POV and Member Selection
- Format member and data cells
- Create a template that can be shared with other users

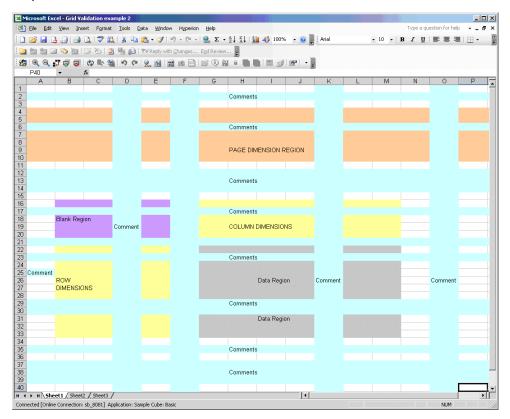
Use VBA functions to customize and automate worksheets

Note:

Smart View recommends a maximum grid size of 65,000 cells.

Smart View Grid Components

The graphic in this section shows the Smart View grid components. In the example Smart View grid, the Row dimensions are to the left and to the bottom of the Column dimensions. The Page dimensions can occur anywhere in the rows above the Column dimensions (applies to Essbase only).



The components of the Smart View grid are described in Table 6.

Table 6 Smart View Grid Components

| Grid Component | Color in Example Grid | Description |
|------------------|-----------------------|--|
| Row Dimension | Yellow | A dimension or dimension members that are placed down one column across one or more rows on a worksheet |
| Column Dimension | Yellow | A dimension or dimension members that are placed on a row across one or multiple columns in a worksheet |
| Page Dimension | Orange | A dimension or dimension member that is placed in the area above the column dimensions in a worksheet (Essbase only) |

| Grid Component | Color in Example Grid | Description |
|----------------|-----------------------|--|
| Comments | Light Blue | Descriptive text added by the user into the worksheet |
| Data Region | Gray | After a grid operation such as a Refresh, Zoom In, or Zoom Out—the area where the row, column, and page data is placed |
| Blank Region | Purple | Areas of the worksheet that are not yet affected by any grid operations |

See "Free-Form Guidelines" on page 121 for some general rules for free-form and descriptions of valid and invalid free-form grid layouts.

See "Answering Business Questions Using Free-Form" on page 139 for an example scenario that incorporates many aspects of free-form reporting.

Free-Form Guidelines

The following guidelines apply when you are working in free-form mode:

- When typing member names that consist of numbers into a blank worksheet, preceded the member name with a single quotation mark. For example, when accessing the Essbase Sample Basic database, type '100 in the worksheet for Product dimension member 100.
- Oracle recommends using Member Selection, as described in "Selecting Members for Ad Hoc Analysis" on page 97, to input member names into a free-form worksheet. Manually typing member names is permitted, but can be error prone.
- Member strings that are admissible in free-form mode are member names and their aliases from the current alias table. Aliases from a noncurrent alias table that are input into a freeform sheet are identified and processed as comment strings.
- When connected to an Essbase or Hyperion Enterprise data source, if you cut and paste from certain sources, such as Microsoft Word, into an Excel worksheet, some of the strings you paste into the grid may be prefixed by hidden characters (control characters). The hidden characters may result in ad hoc or free-form behavior that is misleading or error messages that are not obvious. In such cases, contact your Smart View administrator, who can identify the issue through logs.
- When connected to a duplicate member Essbase data source, be sure to select the Name Only option in the Display tab of the Options dialog box, as described in "Scale and Decimal Display Options" on page 215. Smart View then displays qualified member names in the worksheet.
- When creating or editing a sheet that is connected to a duplicate member Essbase data source, you must use qualified member names for the duplicate members in the sheet.

The following topics contain particular information on working with free-form grids:

- "Working with Attribute Dimensions" on page 122
- "Entering Dynamic Time Series Members" on page 122
- "Submitting Dirty Cells" on page 123
- "Comment Handling" on page 124

- "Preserving Comments, Formulas, and Format" on page 124
- "Handling the #Missing and the #No Access Label" on page 124

For more information about the kinds of grids you can and cannot create with free-form reporting, and on Smart View handling of free-form, see "Valid and Invalid Grids" on page 125.

Working with Attribute Dimensions

In structured grid processing operations, attribute dimensions are not shown. In free-form, you can type an attribute dimension member in the grid and it will be processed and validated. Only the attribute dimension member that you added will be displayed and used during processing and validation; the remaining attribute dimension members will not be included. Structured grid operations that follow this free-form request will retain the attribute dimension member.

Note:

Attribute dimension members are not supported for Hyperion Enterprise.

Entering Dynamic Time Series Members

Smart View supports Dynamic Time Series members typed directly into free-form grids. This section contains some (not all) examples of supported and unsupported formats.

For a complete list of Dynamic Time Series members, see "Working with Dynamic Time Series Members" on page 102.

Note:

Dynamic time series members are not supported for Hyperion Enterprise.

Supported Format Examples

When entering dynamic time series members into a free-form grid, some examples of supported formats are:

- Q-T-D(Jan)
- QTD(Jan)
- Y-T-D(Mar)
- YTD(Mar)
- M-T-D(Jun)
- MTD(Jun)

Unsupported Format Examples

When entering dynamic time series members into a free-form grid, some examples of unsupported formats are:

- Q-T-Date(Jan)
- QTDate(Jan)
- Y-T-Date(Mar)
- Y-T-Date(Mar)
- M-T-Date(Jun)
- MTDate(Jun)

Submitting Dirty Cells

If you modify a data cell in a Smart View grid, Smart View marks the cell "dirty." During a Submit Data operation, dirty cells are updated in the data source.

For Hyperion Enterprise, the default color of dirty cells is yellow.

The Submit Data command is enabled only when you perform either of the following tasks:

- Type over a data cell.
- Select a range of cells and use one of the options in the Adjust Data dialog box to change a data value, as described in "Adjusting and Spreading Data Values" on page 72.

The Submit Data command is disabled in free-form mode when you type over non-data.

In free-form mode, if you want to change and submit new data, refresh the worksheet as described in "Refreshing the Grid" on page 104, modify the data cell; and then select Submit from the menu.

Note:

When the grid layout is changed (such as by inserting a column or row) or when the area outside of the data region is changed, the Submit Data option will be disabled (except Hyperion Enterprise). Select Hyperion, then Refresh to refresh the grid so that the data region can be established by Smart View, then the Submit Data option will be enabled and data can be changed and submitted.

Note:

In Hyperion Enterprise, when you try to submit data after changing the grid layout, you will be prompted to refresh the grid.

See "Submitting Data" on page 114 for information on the Submit command.

Comment Handling

Comments can be placed in various places in the grid. In the example in "Smart View Grid Components" on page 120, you can see comments marked in light blue color.

Comments can be placed as follows:

- Between Row dimensions
- Between Column dimensions
- Between Page dimensions
- Between dimensions and data
- To the left, right, top, bottom of the grid
- Inside the grid range
- Interleaved with members of Page dimensions
- Comment rows and Comment columns can be interleaved with row and columns dimension member
- Interleaved with members of Row, Column and Page dimensions

Comments can be placed anywhere in the grid with the exceptions described in the section, "Valid and Invalid Grids" on page 125.

Preserving Comments, Formulas, and Format

Smart View tries to preserve all comments, formulas, and customized report layouts. Some exceptions that may result in unexpected behavior are when the following actions are performed:

- Zoom in on a Page dimension
- Pivot a dimension from the POV to a row or column
- Drag and drop a dimension from the POV to the worksheet
- Pivot a Row dimension to a Column dimension
- Switch the location of a Row dimension to another row
- Switch the location of a Column dimension to another column
- Change member aliases using the Change Alias Table command

Handling the #Missing and the #No Access Label

Smart View correctly identifies the #Missing label and the #No Access label as data and not as comments. You can set this label in the Display tab of the Options dialog box, using any string you choose, including a blank. In the free- form request, Smart View sends the label string to the data source. If the label is found in member intersection cells in the data region, the cell is correctly identified as data and not as a comment. If this string in a member intersection cell in the data region is anything but the label, Smart View determines the grid to be invalid.

Note:

You can choose to have Smart View display numeric zeroes instead of #Missing or other text. See "Replacement Labels" on page 212.

Valid and Invalid Grids

The following sections contain samples of valid and invalid grid layouts. Use these samples as guidelines when building your own free-form grids. The sample free-form grids pertaining to page dimensions and dynamic time series members are not applicable for Hyperion Enterprise.

For Hyperion Enterprise data sources, you cannot type dimension names in the free-form grids. You can type only member names.

Valid Simple Grid

Figure 6 is a valid simple grid, where Year is the Row dimension, Measures is the Column dimension and Product is the Page dimension. Free form grids do not need to start in cell A1. Blank rows and columns can be inserted above and to the left of the grid. Figure 7 on page 125 is a valid simple grid specific for Hyperion Enterprise.

Figure 6

| | | Α | В | С | D | Е |
|---|---|------|--------|-----------|---------|----------|
| | 1 | | | | Product | |
| | 2 | | Profit | Inventory | Ratios | Measures |
| | 3 | Qtr1 | | | | |
| Г | 4 | Qtr2 | | | | |
| Г | 5 | Qtr3 | | | | |
| | 6 | Qtr4 | | | | |
| | 7 | Year | | | | |

Figure 7

| | Α | В | С | D |
|---|-------|------|------|------|
| 1 | | ORG1 | ORG2 | ORG3 |
| 2 | Q1 08 | | | |
| 3 | Q2 08 | | | |
| 4 | Q3 08 | | | |
| 5 | Q4 08 | | | |

Invalid Grid with Dimension in Both Row and Column Dimensions

Members of a dimension can only be typed in one of the following regions:

- In the same row
- In the same column

Anywhere in the Page dimension region

Figure 8 is invalid because, with Qtr1 typed in cell F2, the Year dimension occurs in both a Row and a Column dimension.

Figure 8

| | Α | В | С | D | Е | F |
|---|------|--------|-----------|--------|----------|------|
| 1 | | | Product | | | |
| 2 | | Profit | Inventory | Ratios | Measures | Qtr1 |
| 3 | Year | | | | | |

Invalid Grid with Dimension in Both Row and Page Dimensions

Figure 9 is invalid because the Year dimension occurs in both a Row and a Page dimension.

Figure 9

| | Α | В | С | D | Е |
|---|------|--------|-----------|--------|----------|
| 1 | | | Product | | Qtr1 |
| 2 | | Profit | Inventory | Ratios | Measures |
| 3 | Year | | | | |

Valid Grid with Different Dimensions in Page Dimension Region

The Page dimension region can contain members of different dimensions. In Figure 10, the Product, Market (member East), Scenario are all Page dimensions.

Figure 10

| | Α | В | С | D | Е |
|---|------|--------|-----------|--------|----------|
| 1 | | | Product | East | Scenario |
| 2 | | Profit | Inventory | Ratios | Measures |
| 3 | Year | | | | |

Invalid Grid with Two Members from Same Dimension in Page Dimension Region

No two members in the Page dimension region can belong to the same dimension. In Figure 11, dimension Market (Market and member East) occurs twice as a Page dimension and the grid is invalid.

Figure 11

| | Α | В | С | D | Е |
|---|------|--------|-----------|--------|----------|
| 1 | | | | East | |
| 2 | | | Market | | |
| 3 | | Profit | Inventory | Ratios | Measures |
| 4 | Year | | | | |

Invalid Grid Where Column Dimension Contains Its Members and a Different Dimension

Every Row and Column dimension can only have members from the same dimension in their rows and columns respectively. Figure 12 is an invalid grid because the Column dimension, Measures, also contains it members in addition to a different dimension, Market. Figure 9 is specific for Hyperion Enterprise. Figure 9 is an invalid grid because the Column dimension (Account), also contains its members in addition to a member from a different dimension (Entity). Smart View interprets the additional Entity dimension (US) as a comment.

Figure 12

| | Α | В | С | D | Е | F |
|---|------|--------|-----------|--------|----------|--------|
| 1 | | | Product | | | |
| 2 | | Profit | Inventory | Ratios | Measures | Market |
| 3 | Year | | | | | |

Figure 13

| | Α | В | С | D |
|---|--------|------|------|----|
| 1 | | ORG1 | ORG2 | US |
| 2 | JAN 07 | | | |
| 3 | FEB 07 | | | |

Valid Grid Where Smart View Interprets Column Dimensions as Page Dimensions

A valid free-form grid must have at least one row and at least one column dimension. However, Figure 14 is an example of a special case. At first glance, it may seem like Figure 14 does not have any Column dimensions and that Measures, Product and Market are all Page dimensions based on the rules stated in the preceding grids. Yet, Smart View validates this grid and interprets it as Row dimension Year, Column dimension Measures, and Page dimension Product and Market.

Where there are no obvious Column dimensions, only one member in only one Row dimension, and multiple members all of different dimensions on the same row and to the top and to the right of the single Row dimension, Smart View interprets the first dimension to be a Column dimension and the dimensions to the right of it as Page dimensions.

Figure 14

| | Α | В | С | D |
|---|------|----------|---------|--------|
| 1 | | Measures | Product | Market |
| 2 | Year | | | |

Invalid Grid with Multiple Members on Same Row With Another Dimension

The grid shown in Figure 15 is invalid because it has both Market and East, both belonging to the same dimension, on the first row along with another dimension, Measures.

Figure 15

| ı | | Α | В | С | D |
|---|---|------|----------|--------|------|
| ı | 1 | | Measures | Market | East |
| ı | 2 | Year | | | |

Invalid Grid Where Smart View Interprets Stacked Dimensions

The first row that has multiple members, all of the same dimension, is identified as a Column dimension. All the dimensions that are placed above this row are candidates for Page dimension, if they comply with the rules for Page dimension. There is an exception to this rule, as shown in Figure 16. According to the rules described to this point, you might expect Product and Market to be interpreted as Page dimensions. However, Smart View interprets Product and Market as Column dimensions, making this grid invalid.

Figure 16

| | Α | В | С | D | Е |
|---|------|---------|-----------|--------|----------|
| 1 | | Market | | | |
| 2 | | Product | | | |
| 3 | | Profit | Inventory | Ratios | Measures |
| 4 | Year | | | | |

The Smart View logic is that if you identify the first Column dimension on row R, and let C be the first column on this row R that contains a member, for each subsequent row above row R, if there is a member on the same column C and that row does not contain any other members, then that row is identified as a Column dimension. Each such row that contains a member that is "stacked" on top of column C is a Column dimension. A row that does not obey this condition is a candidate for a Page dimension as long as it complies with the rules for Page dimensions.

Valid Grid Where Stacked Dimensions Are Interpreted as Page Dimensions

Rows above a page dimension can only be candidates for a Page dimension and cannot be a Column dimension even if they obey the condition for "stackable" members. In Figure 17, Product is a Column dimension since it stacked on top of Profit; Market is a Page dimension since it is not stacked on top of Profit. However, Scenario is a Page dimension, even though it is stacked on top of Profit, because it is a row that is above a Page dimension; hence, it cannot be a Column dimension.

Figure 17

| | Α | В | С | D | Е |
|---|------|----------|-----------|--------|----------|
| 1 | | Scenario | | | |
| 2 | | | Market | | |
| 3 | | Product | | | |
| 4 | | Profit | Inventory | Ratios | Measures |
| 5 | Year | | | | |

Zooming In On a Page Dimension

If you zoom in on a Page dimension, the Page dimension is moved to a Row dimension through a POV to Grid operation.

Figure 18

| Γ | | Α | В | С | D | Е |
|---|---|------|--------|-----------|----------|----------|
| Г | 1 | | | Product | | |
| Г | 2 | | Profit | Inventory | Ratios | Measures |
| | 3 | Year | 105522 | 117405 | 55.26163 | 105522 |

Results of Zoom In

For example, if you zoom in on Market in Figure 18, it will result in Figure 19.

Figure 19

| | | Α | В | С | D | Е | F |
|---|---|---------|------|--------|-----------|----------|----------|
| I | 1 | | | Profit | Inventory | Ratios | Measures |
| | 2 | Product | Year | 105522 | 117405 | 55.26163 | 105522 |

Valid Grid Where Member Overrides Dimension in the Page Dimension Region

If you type a dimension or member name into the Page dimension region, this Page dimension overrides any default or existing POV. For example, in Figure 20, where the user types East into an existing grid which has Product, Market, and Scenario as the POV, then East overrides Market in the POV.

Figure 20

| | Α | В | С | D | Е |
|---|------|--------|-----------|--------|----------|
| 1 | | | | East | |
| 2 | | Profit | Inventory | Ratios | Measures |
| 3 | Year | | | | |

Invalid Grid —Comments in Data Cells

Member intersection cells in the Data region must be data cells and cannot contain comments. Figure 21 is invalid.

Figure 21

| | Α | В |
|---|------|----------|
| 1 | | Measures |
| 2 | Year | Comment |

Invalid Grid Where Blank Region Cells Cannot Be Comments

Member intersection cells in the Blank region must be blank and cannot contain comments. Figure 22 is invalid.

Figure 22

| A | | В | |
|---|---------|----------|--|
| 1 | Comment | Measures | |
| 2 | Year | 105522 | |

Invalid Row and Column Dimensions and First Members

The first members of every column dimension must occur on the same column, and the first members of every row dimension must occur on the same row. Figure 23 is invalid because cell B2 is on the first column of the column dimensions and it has to be a member of the Measures dimension, whereas it is a comment.

Figure 23

| | Α | В | С | D |
|---|------|---------|----------|---------|
| 1 | | 100-10 | 100-30 | 100 |
| 2 | | Comment | Measures | Measure |
| 3 | Year | | | |

Invalid Grid that Contains Comment Row and Comment Column in Dimension Region

Row and column dimension regions can be interleaved with comment rows and comment columns. Figure 24 below is invalid because the comment in cell C2 does not belong to either a comment row or a comment column. (Both row 2 and column C have dimension members.)

Figure 24

| | Α | В | С | D |
|---|------|----------|---------|----------|
| 1 | | 100-10 | 100-30 | 100 |
| 2 | | Measures | Comment | Measures |
| 3 | Year | | | |

Valid Grid Containing Dynamic Time Series Member

Dynamic Time Series (DTS) members can be placed on the free-form grid using Member Selection. Additionally, you can also manually type a DTS member name into a grid. Figure 25 is a valid free-form grid and the member name Q-T-D(Jan) can be manually typed into the grid.

Figure 25

| | А | В |
|---|------------|----------|
| 1 | | Measures |
| 2 | Year | |
| 3 | Qtr1 | |
| 4 | Q-T-D(Jan) | |
| | , , | |

Note:

DTS member formats such as Q-T-D(Jan), QTD (Jan), YTD(Mar), Y-T-D(Mar), MTD (Jun), and M-T-D(Jun) are supported. Formats such as QTDate(Jan) are not supported.

Valid Grid with Duplicate Member Names

In free-form mode, when editing a sheet that is connected to a duplicate member Essbase data source, you must use the qualified member names for the duplicate members in the sheet. Figure 26 shows the syntax to use when typing in duplicate member names.

Figure 26

| | | Α | В |
|---|---|--------------------|----------|
| I | 1 | | New York |
| I | 2 | [Income].[Other] | |
| I | 3 | [Expenses].[Other] | |

Valid Grid That Contains Numeric Member Names

Member and dimension names cannot be comments.

If you want to type a numeric member name using free-form, tag the numeric name as a member by typing a single quote (`) in front of the member name when you enter it in the Excel cell. For example, '100.

By typing the single quote, you differentiate between a numeric member name and a numeric comment or numeric data. In Figure 27, 100 is interpreted as a member.

Figure 27

| | Α | В |
|---|------|------|
| 1 | | Year |
| 2 | '100 | |

Valid Grid With Numeric Comment and Numeric Data

Both numeric comment and numeric data are typed the same by the user. Smart View identifies if it is a numeric comment or numeric data depending on the region in which the number is placed. If it is in the data region, then Smart View identifies it as a data cell; otherwise, it is marked as a comment. In Figure 28, 2005 is identified as data.

Figure 28

| | Α | В |
|---|------|----------|
| 1 | | Measures |
| 2 | Year | 2005 |

Valid Grid Where Numeric Data is Identified as a Comment

In Figure 29, 2005 is identified as a comment since it is not in the data region, and the grid is identified to be valid.

Figure 29

| | Α | В | С |
|---|------|------------|-------|
| 1 | | Report for | Q3 in |
| 2 | | 2005 | |
| 3 | | | |
| 4 | | Measures | |
| 5 | Year | 105522 | |

Valid Grid Consisting of Two Columns by Two Rows

Figure 30 is a basic two columns by two rows layout showing the Product and Market dimensions in the first row and column, and members of Sales and Year in the second row and column. Figure 26 is specific for Hyperion Enterprise. Figure 26 is a basic two columns by two rows layout showing the Scenario and Entity dimensions in the first row and column, and members of Account and Period dimensions in the second row and column.

Figure 30

| | А | В | С | D | Е | F |
|---|--------|------|---------|-----------|---------|----------|
| 1 | | | Product | Product | Product | Product |
| 2 | | | Profit | Inventory | Ratios | Measures |
| 3 | Market | Qtr1 | | | | |
| 4 | Market | Qtr2 | | | | |
| 5 | Market | Qtr3 | | | | |
| 6 | Market | Qtr4 | | | | |
| 7 | Market | Year | | | | |

Figure 31

| | Α | В | С | D | Е | F |
|---|------------|-------|--------|--------|--------|--------|
| 1 | | | Actual | Budget | Daily | Fcst |
| 2 | | | GROUP1 | GROUP3 | GROUP2 | GROUP4 |
| 3 | [ORG].ORG1 | Q1 08 | | | | |
| 4 | USDIV | Q2 08 | | | | |
| 5 | US100 | Q3 08 | | | | |
| 6 | US100A | Q4 08 | | | | |

Valid Grid with Comments Inserted in Blank Row and Column

Figure 32 provides an example of comment usage in Smart View. It is a valid grid with comments bordering the data/metadata region. In this case comments are listed in columns A and H, and in rows 1, 2, and 10. These comments are retained for retrieval and zoom operations. For a list of rules when using comments on a grid, see "Comment Handling" on page 124.

Figure 32

| | Α | В | С | D | Е | F | G | Н |
|----|----|--------|------|---------|-----------|---------|----------|----|
| 1 | | | c1 | d1 | | | | |
| 2 | | | c2 | d2 | | | | |
| 3 | | | | Product | Product | Product | Product | |
| 4 | | | | Profit | Inventory | Ratios | Measures | |
| 5 | a5 | Market | Qtr1 | | | | | h5 |
| 6 | a6 | Market | Qtr2 | | | | | h6 |
| 7 | | Market | Qtr3 | | | | | |
| 8 | | Market | Qtr4 | | | | | |
| 9 | | Market | Year | | | | | |
| 10 | | | c10 | d10 | | | | |

Valid Grid with Formulas Inserted in Blank Row and Column

Figure 33 provides a valid example of Excel-based formula usage in the Smart View environment. Formulas are added to the edges of the grid. These are standard Excel formulas and resolve based on your personal Excel settings. Formulas can be added in any area in which a comment can exist. For a list of formula usage guidelines, see "Working with Formulas in Ad Hoc Grids" on page 114 and "Preserving Comments, Formulas, and Format" on page 124.

Figure 33

| | Α | В | С | D | Е | F |
|---|--------|------|-------------|-----------|---------|-------------|
| 1 | | | Product | Product | Product | Product |
| 2 | | | Profit | Inventory | Ratios | Measures |
| 3 | Market | Qtr1 | | | | |
| 4 | Market | Qtr2 | | | | |
| 5 | Market | Qtr3 | | | | |
| 6 | Market | Qtr4 | | | | |
| 7 | Market | Year | | | | |
| 8 | | | =SUM(C3:C6) | | | =SUM(F3:F6) |

Valid Grid with POV Region and Attribute

Figure 34 provides a valid example of both the page region and attribute usage. In this example, Pkg Type and Budget are understood (by Smart View) to be page dimensions. Additionally, Pkg Type is an attribute dimension attached to the base member product. By drilling down on Pkg Type you can do attribute based analysis on measures as it relates to specific Product attributes. This can be further used to create a cross-tab analysis of product SKUs by attribute. For a description of zoom behavior when working with the page area see the description on grid "Zooming In on Dimension Members" on page 104 and "Zooming Out on Dimension Members" on page 105.

Figure 34

| | Α | В | С | D | Е | F |
|----|----------|------|---------|-----------|---------|----------|
| 1 | Pkg Type | | | | | |
| 2 | Budget | | | | | |
| 3 | | | | | | |
| 4 | | | Product | Product | Product | Product |
| 5 | | | Profit | Inventory | Ratios | Measures |
| 6 | Market | Qtr1 | | | | |
| 7 | Market | Qtr2 | | | | |
| 8 | Market | Qtr3 | | | | |
| 9 | Market | Qtr4 | | | | |
| 10 | Market | Year | | | | |

Valid Grid with Complex Comments

Figure 35 provides a valid, combined sample of using the Page region, attributes, and comments on a single grid.

Figure 35

| | А | В | С | D | Е | F | G | Н |
|----|----------|----|------|---------|-----------|-----|---------|----------|
| 1 | Pkg Type | | | | | | | |
| 2 | Budget | | | | | | | |
| 3 | | ВЗ | C3 | D3 | | | | |
| 4 | | | | Product | Product | | Product | Product |
| 5 | | | | | E5 | F5 | G5 | |
| 6 | | | | Profit | Inventory | | Ratios | Measures |
| 7 | Market | | Qtr1 | | | | | |
| 8 | Market | | Qtr2 | | | | | |
| 9 | | В9 | | | E9 | F9 | G9 | |
| 10 | Market | | Qtr3 | | | | | |
| 11 | Market | | Qtr4 | | | | | |
| 12 | Market | | Year | | | | | |
| 13 | | | | | | | | |
| 14 | | | | D14 | E14 | F14 | | |

Retrieving in Free-Form Mode

- To construct a free-form report:
- 1 Open a worksheet and connect to a data source.
- 2 In the worksheet, type in the member names for the report in the layout that you want to create.

For example, in a blank worksheet connected to the Sample Basic application and database in Essbase, to see the total product sales in New York, you may type New York in cell B1 and Sales in cell A2.

If a member name you entered is a duplicate, the Member Name Resolution dialog box is displayed. See "Resolving Member Names in Free-Form Grids" on page 138 for information on using this dialog box.

Note:

If a member name consists of a number, such as 100, you must precede the member name with a single quotation mark (for example, '100). This rule also applies to member names with spaces between words.

Refresh the grid using any of the methods described in "Refreshing the Grid" on page 104.

Smart View retrieves data from the connected data source for the members you entered into the free-form report.

Optional: Perform further Ad Hoc operations on the worksheet.

For example, you may proceed by performing one or more of the following operations:

- Zoom in on dimensions and members, as described in "Zooming In on Dimension Members" on page 104.
- Insert rows and add formulas to cells, as described in "Working with Formulas in Ad Hoc Grids" on page 114.
- Pivot dimensions from the POV to the worksheet, as described in "Pivoting Dimensions" on page 108.

See "Answering Business Questions Using Free-Form" on page 139 for an example free-form scenario.

Resolving Dimension Names

When you select Hyperion, then Member Selection in a blank grid, the Dimension Name Resolution dialog box is displayed, where you choose a dimension or members to place on the grid. This process is referred to as resolving dimensions because when you choose the Member Selection option on a blank sheet, the data source has no way of predicting which dimension or members, including attribute members, you want to see and choose from in the Member Selection dialog box.

When you choose dimensions and members using this method, be sure that the first cell you select on the sheet is the cell in which you want to begin your free-form layout.

Note:

For Hyperion Enterprise data source, you cannot place dimensions on the grid.

- To resolve dimensions names in a grid:
- 1 In a blank worksheet, select a cell.

For example, to build a grid that starts from the left column, select cell A2. To build a grid that starts on the first row, select cell B1.

- 2 Select Hyperion, then Member Selection. For Office 2007, click Member Selection in the Task section of the Hyperion ribbon.
- In the **Dimension Name Resolution** dialog box, select the dimension to place on the sheet.

For example, if you are connected to an Essbase data source, select the Product dimension.

To orient members vertically in the worksheet starting from the cell you selected in step 1, select the Vertical Orientation check box.

This check box is cleared by default; meaning that members will be oriented horizontally across the sheet from the cell you selected in step 1.

Click **OK** to launch the **Member Selection** dialog box.

- 6 Select the members to place on the worksheet, as described in "Selecting Members for Ad Hoc Analysis" on page 97, and click OK.
- 7 View the layout of the dimension or members you just placed on the sheet.
- Perform one of the following actions:
 - If you placed members vertically on the grid:
 - Repeat step 1 through step 7 to place a new set of members horizontally on the grid
 - Use the POV to pivot a dimension to a row
 - Type a dimension or member name (including attribute dimension or member names) directly on the grid
 - If you placed members horizontally on the grid:
 - Repeat step 1 through step 7 to place a new set of members vertically on the grid
 - Use the POV to pivot a dimension to a column
 - Type a dimension or member name (including attribute dimension or member names) directly on the grid
- Refresh the grid using any of the methods described in "Refreshing the Grid" on page 104.

Retrieving Attribute Dimensions in Free-Form Mode

You retrieve attribute dimensions using a method similar to that described in "Resolving Dimension Names" on page 136.

You can also retrieve an attribute member by typing the name directly in the worksheet, as described in "Retrieving Attribute Dimensions and Members" on page 105. This method can be used when the base dimension already exists on the worksheet.

The method for retrieving attributes described in this section assumes you are starting with a blank worksheet.

Note:

Hyperion Enterprise does not support Attribute dimensions.

- To retrieve an attribute dimension in free-form using Member Selection:
- 1 In a blank worksheet, select a cell.

For example, to build a grid from the left column, select cell A2. To build a grid that starts on the first row, select cell B1.

- 2 Select Hyperion, then Member Selection. For Office 2007, click Member Selection in the Task section of the Hyperion ribbon.
- 3 In the **Dimension Name Resolution** dialog box, select the attribute dimension.

For example, if you are connected to an Essbase data source, to query by package type, select the Pkg Type attribute dimension.

4 To orient members vertically in the worksheet starting from the cell you selected in step 1, select the Vertical Orientation check box.

This check box is cleared by default; meaning that members will be oriented horizontally across the sheet from the cell you selected in step 1.

- 5 Click **OK** to launch the **Member Selection** dialog box.
- 6 Select the members to place on the worksheet, as described in "Selecting Members for Ad Hoc Analysis" on page 97.

Note:

You can also add attribute dimensions and members to the sheet using the procedure in "Retrieving Attribute Dimensions and Members" on page 105.

Resolving Member Names in Free-Form Grids

When working in a free-form grid, Smart View may not be able to identify the dimension to which a member that you have typed belongs. For example, if you type the member name "Albany" on an empty worksheet, Smart View cannot know whether this signifies the member Albany whose parent is New York or California, or whether the member Albany comes from a market or a customer dimension. In this case, Smart View asks you to resolve the member by providing the dimension to which the member belongs.

When you type a member name that Smart View cannot resolve, the Member Name Resolution dialog box is displayed.

Note:

The Member Name Resolution dialog box only appears when you are connected to a Financial Management or Hyperion Enterprise data source. If you are connected to an Essbase data source, an error message is displayed. In this case, use Member Selection to choose member names, as described in "Selecting Members for Ad Hoc Analysis" on page 97.

- To resolve a member name:
- 1 In the Member Name Resolution dialog box, select the dimension to which the member belongs from the drop-down list.

If a second unresolved member exists, a second drop-down list is displayed.

- 2 Optional: If a second unresolved member exists, select the dimension to which the member belongs from the second drop-down list.
- 3 Click OK.
- 4 Refresh the grid using any of the methods described in "Refreshing the Grid" on page 104.

Smart View retrieves data from the connected data source and displays the resolved members in the following format: [Dimension]. [Member]. This format is not supported for Hyperion Enterprise data sources. For Hyperion Enterprise, Smart View displays the resolved member

names in the following format: Member. The number of members displayed in the name depends on the number of members necessary to uniquely identify the member.

Answering Business Questions Using Free-Form

Free-Form reporting gives you the ability to answer business questions beyond those represented on a standard report. By providing the ability to quickly create an ad hoc grid, through typing or using member selection, Free-Form reporting moves you from question to answer. For example, what if you wanted to know: What are my actual sales in New York?

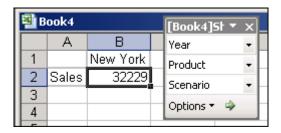
The following steps lead you through a series of questions based on the Essbase Sample Basic database. The process starts with a single business and culminates with a reusable report template.

Note:

The scenario in this topic assumes that you are connected to the Essbase Sample Basic database.

- To construct an example free-form report:
- 1 Type the member names Sales in cell A2 and New York in cell B1, and then select Hyperion, then Refresh. For Office 2007, click Refresh in the Review section of the Hyperion ribbon.

You should see a grid similar to this:



The total Sales figure for New York is \$32,229. But how does this figure break down? This leads to new questions, such as what are the sales by product? What are the sales by quarter?

- 2 View sales by product using the Select Member option in the POV to select all the products from Product dimension.
 - In the POV, click Options, then Select Members, then Product.
 - In the Select Member dialog box, select Product and members 100, 200, 300, and 400 and then click
 - Click OK. c.

The POV should look similar to this:



3 Click the arrow in the Product dimension, select 100, and refresh the grid to view only the sales total for product 100.

Refresh the grid by selecting Hyperion, then Refresh; clicking , or clicking the green arrow in the POV. For Office 2007, click Refresh in the Review section of the Hyperion ribbon.

- 4 In the POV, click the arrow next to 100 and select 200, then refresh the grid to view the sales total for product 200; repeat to view sales totals for products 300 and 400.
- 5 Click the arrow next to 400 and select Product; and then drag Product from the POV to the grid and drop it on the Sales cell.
- Click Options or select Hyperion, then Options, and in the Options dialog box, set the following options:
 - In the Ad Hoc tab, select Suppress Repeated Members options.
 - In the Display tab, clear the #No Data/Missing Label text box.

Refresh the grid to view the effect of these options changes.

Note:

The Suppress Repeated Members option only affects member names in the outermost row or columns.

Change the nesting order by moving Sales; right-click Sales and drag it on top of the Product cell.

Note:

When Sales was moved to the outer column in the grid, the repeated members in the column were suppressed.

8 Pivot the dimensions on the grid by selecting either the Sales or Product dimension and clicking 🗊 or selecting Hyperion, then Ad Hoc Analysis, and then Pivot; try this several times to see different layouts of the grid.

When there are two row dimensions, pivot one of the row dimensions. When there are two column dimensions, pivot one of the column dimensions.

9 Click until the grid appears as it was in step 7.

Next, we will take a look at how New York breaks down across product lines and scenarios.

10 Pivot the Scenario dimension to a column.

In the POV, click Options, then Pivot to Column, and then Scenario to place the Scenario dimension on the worksheet. The Scenario dimension should appear above New York in the grid.

11 In the worksheet, zoom in on Scenario.

Select Scenario and zoom in by performing one of the following actions:

- Select Hyperion, then Ad Hoc Analysis, and then Zoom In. For Office 2007, click Zoom In in the As Hoc Analysis section of the Hyperion ribbon.
- Click 🔎
- Double-click Scenario.

12 Change the replacement text to display for missing or no label cells to a hyphen (-).

- Click Options or select Hyperion, then Options, and then click Display. a.
- b. In the #NoData/Missing Label text box, type a hyphen, -, then click OK.
- c. Refresh the grid to view the changes in the Diet row.

Remove the Scenario column.

After the zoom in operation, we decide that the Scenario column is not really needed. To remove the Scenario column:

- Select the cell that contains "Scenario." This should be cell G1, but may be different in your worksheet.
- Click Remove Only or select Hyperion, then Ad Hoc Analysis, and then Remove Only. b.

Move Product to the bottom of the grid.

Perform the following steps to move the Product dimension below the product members in the worksheet:

- Zoom out on member 100 by selecting 100 in the grid and clicking . a.
- Delete row 4, the row that is the result of the zoom out operation. Select the row and use Ъ. the Excel command, Edit, then Delete, to delete the row.
- Zoom in on the Product dimension to expand it and place Product at the bottom of the c. grid, as shown:

Now we would like to see a quarterly breakdown.

15 Pivot the Year dimension to a row.

In the POV, click Options, then Pivot to Row, and then Year to place the Year dimension on the worksheet. The Year dimension should appear to the left of the Sales dimension in the grid.

Now we want to see a quarterly breakdown.

16 Zoom in on the Year dimension.

Zoom in by performing one of the following actions:

- Select Year and then select Hyperion, then Ad Hoc Analysis, and then Zoom In.
- Click 🔑
- Double-click on Year.

The grid is getting long. We will shorten the length.

Perform the following steps to shorten the grid length:

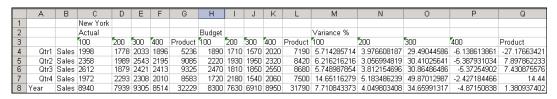
- Drag 100 in cell C3 on top of New York in cell D2.
- b. Next, drag New York in cell C3 on top of Actual in cell C1.

Now the grid is wide. We will shorten the width.

Perform the following steps to remove the Variance and Diet members from the grid.

- Select the cell that contains "Variance." This should be cell O2, but may be different in your worksheet.
- Select Hyperion, then Ad Hoc Analysis, and then Remove Only or click let to remove the b. entire column and associated columns from the grid.
- Select the cell that contains "Diet." This should be cell S3, but may be different in your worksheet.
- d. Repeat step 18.b.

The grid should look similar to this:



After a review of the worksheet, we would like to add first and second half totals to the grid, as well a variance for those totals. In the next series of steps, we will add several formulas to the worksheet.

19 Add three new rows at the bottom of the worksheet, above Year.

You should have three blank rows between Q4 and Year.

20 Type labels for the new rows.

- a. In cell A8, type First Half Total.
- b. In cell A9, type Second Half Total.
- c. In cell A10, type First/Second Variance.

21 Use Excel's formula functionality to add formulas for the first half and second half totals.

In the first new row (row 8), enter a formula in the 100 column under Actual to total Q1 and Q2.

For example, in cell C8, enter the formula:

=SUM(C4:C5)

Then, copy the formula across the row from cell D8 to cell L8.

b. In the second new row (row 9), enter a formula in the 100 column under Actual to total Q3 and Q4.

For example, in cell C9, enter the formula:

=SUM(C6:C7)

Then, copy the formula across the row from cell D9 to cell L9.

22 Use Excel's formula functionality to add a formula to figure the difference between first half and second half sales totals.

In the third new row (row 10), add a formula in the 100 column under Actual.

For example, in cell C10, enter the formula:

```
=SUM(C8-C9)
```

Then, copy the formula across the row from cell D10 to cell L10.

- 23 Use Excel's formula functionality to add formulas to average the variance figures for the first half and second half.
 - In cell M8 of the first new blank row, enter a formula in the 100 column under Variance % a. to average the Q1 and Q2 variance percentages.

For example, in cell M8, enter the formula:

```
=AVERAGE (M4:M5)
```

Then, copy the formula across the row from cell N8 to cell Q8.

In cell M9 of the second new blank row, enter a formula in the 100 column under b. Variance % to average the Q3 and Q4 variance percentages.

For example, in cell M9, enter the formula:

```
=AVERAGE (M6:M7)
```

Then, copy the formula across the row from cell N9 to cell Q9.

24 Use Excel's formula functionality to add a formula to figure the difference between the first half and second half average variance figures.

In cell M10 of the third new blank row, add a formula in the 100 column under Variance % to figure the difference between the first half and second half variance percentages.

For example, in cell M10, enter the formula:

```
=SUM(M8-M9)
```

Then, copy the formula across the from cell N10 to cell Q10.

25 Use Excel's formula functionality to add formulas to average the variance figures for the first half and second half.

In cell M11, on the same row as Year, add a formula in the 100 column under Variance % to average the first half and second variance percentages.

For example, in cell M11, enter the formula:

```
=AVERAGE (M8:M9)
```

Then, copy the formula across the row from cell N11 to cell Q11.

Now we can create a template from our worksheet. By doing so, this worksheet can be shared with other users. First, we will do some further formatting of the sheet.

26 Select Hyperion, then Options, then Display tab and select the Use Excel Formatting check box.

Alternatively, you may choose to clear the Use Excel Formatting check box and use the Capture Formatting command to retain the formatting styles that you apply, or use Cell Styles to apply styles based on intersections in the connected database. However, certain Excel styles you apply using Capture Formatting or Cell Styles may be lost upon refresh.

27 Decrease the decimal places of the variance percentage figures.

In Excel, highlight the range of variance percentage figures, select Format, then Cells, and for the Number category on the Number tab, decrease the decimal places to 2.

28 Add dollar signs and decimal places to the Sales figures.

In Excel, highlight the range of sales figures, select Format, then Cells, and for the Currency category on the Number tab, increase the decimal places to 2 and choose the dollar sign (\$) as the currency symbol.

29 Use Excel formatting to add background colors to the grid.

Alternatively, you may choose to use the Capture Formatting command to retain the colors after a refresh.

For example, select cells C4 through F7, select a fill color from the Excel toolbar, then select Hyperion, then Capture Formatting. Be sure the range of cells is still highlighted when you select the Capture Formatting command.

Capture Formatting only retains formats on data cells that were derived from your ad hoc operations. It does not retain formats for the cells that were manually added to the grid.

30 Display aliases for member names in the worksheet.

Select Hyperion, then Ad Hoc Analysis, and then Change Alias Table and, for this example, choose "Default" in the Select Alias Table dialog box.

For more information on using aliases for member names, see "Displaying Aliases for Member Names" on page 107.

31 Save the workbook.

32 Using VBA functionality, add buttons to the worksheet.

To simplify connecting and refreshing for users with whom you may share this worksheet, use Excel's Visual Basic Editor to add buttons to the worksheet and use the Smart View VBA code samples listed to attach functionality to each button:

- Connect button, to make a data source connection
 - x = HypMenuVConnect
- Refresh button, to perform a refresh of the worksheet
 - x = HypMenuVRefresh
- Disconnect button, to disconnect from the data source
 - x = HypMenuVDisconnect

Instructions for using VBA functions are in Chapter 19, "VBA Functions."

See "Using Smart View VBA Functions" on page 232 for instructions on adding a button and assigning a VBA macro to it.

33 Refresh the worksheet and review the formatting and layout.

For example, check the following items:

- Review the formatting of the sheet. Has Excel formatting you applied been retained? Note that selections made in the Cell Styles tab of the Options dialog box are overridden if you are using Excel formatting.
- Test the buttons you added in step 32. Is the correct functionality attached to the button?
- Are the Excel formulas you entered are still valid?

Figure 36 is an example of the spreadsheet developed in this section, with the exception of the background color formatting and the placement on the worksheet. Also, a title was added to the page and the cells surrounding the active grid were filled with white using the Fill Color button on Excel's Formatting toolbar.

This worksheet provides information to users about the sales of product in New York. It provides easy data source connection access and refresh functionality. It can be reused by other regional managers, for example, by replacing the state name, New York, with another state name, like Florida.

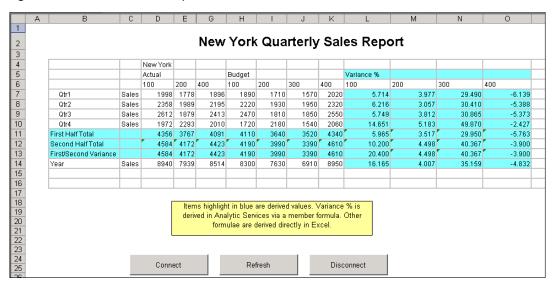


Figure 36 Formatted Free-Form Report

Constructing an Example Free-form Report Using a Hyperion Enterprise Data Source

Consider an application in Hyperion Enterprise which has a schedule with the following POV (Point Of View):

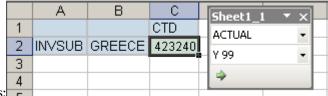
- Category-Actual
- Period-Jan 99
- Organization-Europe

- **Entity-Greece**
- Account-INVSUB (Investments in Subsidiaries)

The scenario in this section assumes that you are connected to this application and the following steps lead you through a series of questions based on this application. For example, what if you wanted to know: What are my investments in subsidiarires of Greece?

The process starts with a single business and culminates with a reusable report template.

- To construct an example free-form report:
- 1 Type the member names INVSUB in cell A2, GREECE in cell B2, and CTD in cell C1. Select Hyperion, then Refresh.



You should see a grid similar to this:

The total investments in subsidiarires for New York is \$423240. But how does this figure break down? This leads to new questions, such as what are the investments by each subsidiary? What are the investments by quarter?

- View investments by each subsidiary using Member Selection to select all the subsidiarires from Account dimension.
 - Click cell A2 and select Hyperion, then Member Selection. a.
 - b. In Member Selection, select Account and children members of INVSUB and then click
 - Click OK. c.
- Select **Hyperion, then Options** and in the **Options** dialog box, set the following options:
 - In the Ad Hoc tab, select Suppress Repeated Members options.
 - In the Display tab, clear the #No Data/Missing Label text box.

Refresh the grid to view the effect of these options changes.

Note:

The Suppress Repeated Members option only affects member names in the outermost row or columns.

Change the nesting order by moving GREECE; right-click GREECE and drag it on top of the INVSUB cell.

Note:

When GREECE was moved to the outer column in the grid, the repeated members in the column were suppressed.

5 Pivot the dimensions on the grid by selecting either the GREECE or INVSUB member and clicking or selecting Hyperion, then Ad Hoc Analysis, then Pivot; try this several times to see different layouts of the grid.

When there are two row dimensions, pivot one of the row dimensions. When there are two column dimensions, pivot one of the column dimensions.

- 6 Click the **Undo** button, , until the grid appears as it was in step 7.
- 7 Pivot the member Y 99 to a column.

In the POV, right-click, drag and drop Y 99 on top of cell C1. The Y 99 member should appear above CTD in the grid.

In the worksheet, zoom in on Y 99.

Select Y 99 and zoom in by performing one of the following actions:

- Select Hyperion, then Ad Hoc Analysis, then Zoom In.
- Click 🔎
- Double-click on Y 99.
- Change the replacement text to display for missing or no label cells to a hyphen (-).
 - Select Hyperion, then Options, and then click Display.
 - b. In #NoData/Missing Label, type a hyphen, -, then click OK.
 - Refresh the grid to view the changes in the INVSUB.SPAIN row.
- 10 Remove the Y 99 column.

After the zoom in operation, we decide that the Y 99 column is not really needed. To remove the Y 99 column:

- Select the cell that contains "Y 99." This should be cell C1, but may be different in your worksheet.
- Select Hyperion, then Ad Hoc Analysis, and then Remove Only or click . The entire column is removed from the grid.
- 11 Move INVSUB to the bottom of the grid.

Perform the following steps to move the INVSUB member below the children members in the worksheet:

- Zoom out on member INVSUB.ITALY by selecting it in the grid and clicking a.
- Select Hyperion, then Options and click Ad Hoc. In Ancestor Position, select Bottom and b. click OK.
- Zoom in on the INVSUB member to expand and place it at the bottom of the grid. c. Now we would like to see a quarterly breakdown.
- 12 Pivot Year dimension members to a row.

Select HY1 99 in the grid and select Hyperion, then Ad Hoc Analysis, and then Pivot.

13 Zoom in on HY1 99 member.

Zoom in by performing one of the following actions:

- Select HY1 99 and then select Hyperion, then Ad Hoc Analysis, and then Zoom In.
- Click 🔎
- Double-click on HY1 99.
- 14 Repeat step 12 for HY2 99. Remove HY1 99 and HY2 99 rows by selecting it in the grid and selecting Hyperion, then Ad Hoc Analysis, and then Remove Only.

The grid getting long, we will shorten the length.

15 Drag INVSUB.ITALY in cell C2 on top of CTD in cell D1 to shorten the grid length.

Now the grid is wide. We will shorten the length.

- 16 Perform the following steps to remove the INVSUB.SPAIN member from the grid.
 - Select the cell that contains "INVSUB.SPAIN." a.
 - Select Hyperion, then Ad Hoc Analysis, and then Remove Only or click uto remove b. the entire column and associated columns from the grid.

After a review of the worksheet, we would like to add first and second half totals to the grid, as well a variance for those totals. In the next series of steps, we will add formulae to the worksheet.

17 Add three new rows at the bottom of the worksheet.

You should have three blank rows after Q4 99.

- 18 Type labels for the new rows.
 - In cell A7, type First Half Total.
 - b. In cell A8, type Second Half Total.
 - In cell A9, type First/Second Variance. c.
- 19 Use Excel's formula functionality to add formulas for the first half and second half totals.
 - In the first new row (row 8), enter a formula in the INVSUB.ITALY column to total Q1 99 and O2 99.

For example, in cell C7, enter the formula:

```
=SUM(C3:C4)
```

Then, copy the formula across the row from cell D7 to cell G7.

In the second new row (row 9), enter a formula in the INVSUB.ITALY column to total Q3 99 and Q4 99.

For example, in cell C8, enter the formula:

```
=SUM(C5:C6)
```

Then, copy the formula across the row from cell D8 to cell G8.

20 Use Excel's formula functionality to add a formula to figure the difference between first half and second half sales totals.

In the third new row (row 9), add a formula in the INVSUB column.

For example, in cell C9, enter the formula:

```
=SUM(C7-C8)
```

Then, copy the formula across the row from cell D9 to cell G9.

21 Select Hyperion, then Options, then Display and select the Use Excel Formatting check box.

Alternatively, you may choose to clear the Use Excel Formatting check box and use the Capture Formatting command to retain the formatting styles that you apply, or use Cell Styles to apply styles based on intersections in the connected database. However, certain Excel styles you apply using Capture Formatting or Cell Styles may be lost upon refresh.

22 Add dollar signs and decimal places to the figures in data region cells.

In Excel, highlight the range of data region cells, select Format, then Cells, and for the Currency category on the Number tab, increase the decimal places to 2 and choose the dollar sign (\$) as the currency symbol.

23 Use Excel formatting to add background colors to the grid.

Alternatively, you may choose to use the Capture Formatting command to retain the colors after a refresh.

For example, select cells C7 through G9, select a fill color from the Excel toolbar, then select Hyperion, then Capture Formatting. Be sure the range of cells is still highlighted when you select the Capture Formatting command.

Capture Formatting only retains formats on data cells that were derived from your ad hoc operations. It does not retain formats for the cells that were manually added to the grid.

24 Save the workbook.

25 Refresh the worksheet and review the formatting and layout.

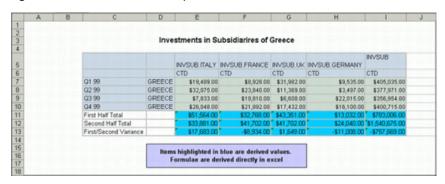
For example, check the following items:

- Review the formatting of the sheet. Has Excel formatting you applied been retained? Note that selections made in the Cell Styles tab of the Options dialog box are overridden if you are using Excel formatting.
- Are the Excel formulas you entered are still valid?

Figure 24 is an example of the spreadsheet developed in this section, with the exception of the background color formatting and the placement on the worksheet. Also, a title was added to the page and the cells surrounding the active grid were filled with white using the Fill Color button on Excel's Formatting toolbar.

This worksheet provides information to users about the investments in subsidiarires in GREECE. It provides easy data source connection access and refresh functionality. It can be reused by other regional managers, for example, by replacing the state name, GREECE, with another name, like New York.

Figure 37 Formatted Free-Form Report



Analyzing Time-Related Data in Free-Form Mode

Data Sources: Essbase

Using Smart View, you can analyze flash metrics such as sales of cost of goods sold against timebased metrics. This enables you to look for trends, find averages for different time periods, and so forth. To do this, you use linked attributes which enable periodicity of members. Periodicity is a shared pattern among time dimension members that make them meaningful for time-based analysis (January and April share periodicity as first months of quarters, for example). Day by month, day by week, and week by year are examples of linked attributes. You can also set ranges for linked attributes and apply filters.

- To analyze time-related data in free-form mode:
- 1 Open an Excel worksheet and connect to a data source.
- 2 Select Date-Time member and select Hyperion, then Member Selection to open Member Selection, where you can select members and apply filters as described in "Filtering Members in Queries" on page 90. For Office 2007, click Member Selection in the Task section of the Hyperion ribbon.
- To enter linked attributes, insert a new column to the left of the Date-Time member and select Hyperion, then Member Selection. Linked attributes are selectable from the Dimensions dropdown menu in the **Dimension Name Resolution** dialog box.
- Select Hyperion, then Refresh. For Office 2007, click Refresh in the Review section of the Hyperion ribbon.

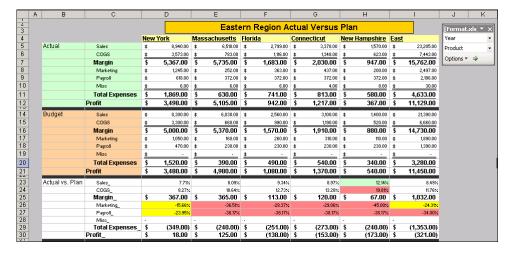
Highly Formatted Free-Form Report Example

You can use Free-Form to build and customize a report to your own specifications, using the formatting options you require. Figure 38 is an example of a highly formatted free-form report. This report example was created using the following Excel formats:

- Standardized column widths
- Cell fill (using multiple colors for each scenario, including dark line breaks)
- Cell-based double-underscoring above totals (for example, row 6)
- Currency and percentage symbols added in various columns

- Excel formulas to derive additional values (rows 23-30)
- Excel Conditional formatting for dynamic highlighting (for example rows 26 and 27)
- Text box with stylized text for the grid label

Figure 38 Highly Formatted Free-Form Report



Creating and Distributing Report Templates Using Free-Form

Free-form reporting provides a solid foundation for the creation and distribution of report templates. For the scenario in this section, assume that part of your job was creating a report such as that shown in "Highly Formatted Free-Form Report Example" on page 150. Also assume that you want to create a series of product line detail reports across multiple markets, based on the highly-formatted report. Furthermore, the reporting requirements span data sources in both Essbase and Financial Management. You can use the free-form capabilities of Smart View to create your report templates.

The following steps use the sample applications provided with Essbase. Note that these steps work for Financial Management or Hyperion Enterprise or Essbase data sources.

These sections comprise the taskflow for report template creation and distribution:

- "Creating a Base Grid Using Dimension Name Resolution" on page 152
- "Using Free-Form to Expand the Layout" on page 152
- "Leveraging Excel Formulas" on page 153
- "Formatting the Grid" on page 153
- "Using the POV as a Page Selector" on page 154
- "Creating Additional Reports" on page 154

Creating a Base Grid Using Dimension Name Resolution

With Smart View, you can select members into empty cells in a grid. This process is referred to as dimension name resolution and allows you to select members without typing or using the POV member selector. Additionally, you can use this option to lay out a simple grid as a foundation for analysis.

Note:

Before beginning the procedure, ensure that Indentation option is set to "Subitems," as described in "Indenting Member Names" on page 208.

- To create a base grid using dimension name resolution:
- 1 On a blank Excel worksheet, connect to the Essbase Sample Basic data source.
- 2 Place the cursor in cell C5 and select Hyperion, then Member Selection to display the Dimension Name Resolution dialog box. For Office 2007, click Member Selection in the Task section of the Hyperion ribbon.
- 3 Select Measures from the list, and select Vertical Orientation, and then click OK.

Selecting Vertical Orientation places the resulting members vertically onto the grid, whereas clearing Vertical Orientation places the resulting members horizontally onto the grid.

- 4 In Member Selection, from the Members list, select these members, and click OK:
 - Profit Total Expenses
 - Margin Marketing
 - COGS Payroll
 - Sales Misc
- 5 Place the cursor in D4 and select Hyperion, then Member Selection.
- 6 In Dimension Name Resolution, select Product from the list; do not select Vertical Orientation.
- 7 In Member Selection, from the Members list, select one product line and its children; for example, select product line 100 and its children.
- 8 Refresh the worksheet.

Using Free-Form to Expand the Layout

You can useSmart View free-form capabilities to further expand the grid by:

- Typing directly onto the grid
- Using the Microsoft Office copy and paste functionality
- To make further member selections using free-form capabilities:
- 1 Select cell B5 and type Actual.
- 2 Refresh the worksheet.

Notice that Scenario is removed from the POV because a member from the Scenario dimension, Actual, is now represented on the grid.

- 3 Select cells C5 through C12, copy them, and then paste them starting in cell C13.
- 4 In cell B13, type Budget.
- 5 Refresh the worksheet.

Leveraging Excel Formulas

Smart View provides the capability of preserving Excel-based formulas entered on a grid. These can be used to further extend the analysis begun with an ad hoc query. Assume for this example that you want to determine budget variances in Excel.

- To leverage Excel formulas in the worksheet:
- 1 Select cells C5 through C12, copy them, and then paste them starting in cell C21.
- 2 In cells C21 to C28, type an underscore (_) at the end of each member name; for example, Profit_. Adding the underscore character changes the cell from a member cell to a label cell. This allows you to use labels for the budget variance rows that are almost the same as the member cells.

Tip:

The underscore is suggested because it is fairly imperceptible to users, but Smart View treats it as a label.

- 3 In cell B21, type Actual vs. Plan.
- Refresh the grid to validate that the labels in the variance area remain undisturbed.
- 5 In cell D21, enter this formula:

```
=D5-D13
```

6 Fill the formula down and across all of the cells in the variance region, and refresh the worksheet. After refresh, the formulas are preserved.

Formatting the Grid

Smart View provides a myriad of formatting options. You can take advantage of the Smart View capture formatting feature, Hyperion product-specific styles, and Excel-based formatting. This example demonstrates applying Excel-based formatting to the grid.

- To apply Excel-based formatting to the grid:
- 1 Select Hyperion, then Options to display the Options dialog box. For Office 2007, click Options in the **Options** section of the **Hyperion** ribbon.
- 2 Select the Display tab, select Use Excel Formatting, and click OK.
- 3 Apply these Excel-based formats:

- Fill the worksheet, outside of the data region light grey.
- Fill product cells in row 4 with light yellow •
- Fill cells B5 through C12 with light green
- Fill cells B13 through C20 with light blue
- Fill cells B21 through C28 with white
- Place double-underlines beneath all total accounts, and place these in a larger, bold font
- Insert a row, filled with black, in between each scenario region
- Make all column widths 20
- Place currency symbols on all values
- Using a text box, place a title on the report
- Refresh the worksheet and verify that the formats and formulas are preserved.

Using the POV as a Page Selector

Once you have a formatted grid, you can select members from dimensions in the POV to quickly view the worksheet from the point of view of the member selected and still preserve overall grid formatting. For example, you may select the region members from the Market dimension, then select a specific region from the POV. After refreshing the grid, your formulas and data points are updated to reflect the point of view of the selected region, preserving any grid formatting you have applied.

- To use the POV as a page selector:
- 1 View sales by market using the Select Member option in the POV to select all products from all regions in the Market dimension.
 - In the POV, click Options, then Select Members, then Market. a.
 - b. In Member Selection, select Market and members East, West, South, and Central, and then click
 - c. Click OK.
- 2 Click the arrow in the Market dimension, select East, and refresh the grid to view only the totals for the East region.
- 3 In the POV, click the arrow next to East and select West, then refresh the grid to view the totals for the West region; repeat to view totals for the South and Central regions.

Creating Additional Reports

Once you have a report that contains, for example, the formatting and formulas you want, you can use this report as a template for creating additional reports.

For example, assume you want a product detail report for each product line. Using the free-form capabilities previously described, you can create subsequent reports for product lines 200 through 400.

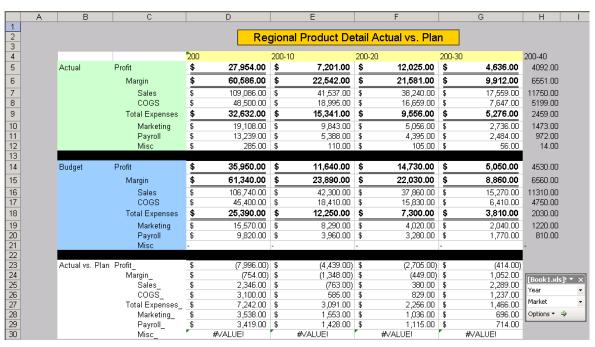
- To create additional reports using the original report as a template:
- 1 Ensure that the Excel workbook contains five worksheets.
- 2 Copy the contents of Sheet 1 onto Sheet 2.
- 3 On Sheet 2, select product 100 in cell D4.
- 4 Select Hyperion, then Member Selection to display the Member Selection dialog box.
- In the **Members** area, select product 200 and its children members, and then add them to the **Selection** area.
- 6 If product 100 is in the selection area, remove it.

The Selection area should contain the members:

- 200
- 200-10
- 200-20
- 200-30
- 200-40

7 Refresh the worksheet.

Notice that after the new members are on the grid, one column extends past the previously formatted region:



8 To format the new column to match the existing columns, apply Excel formatting.

Repeat step 3 through step 8 for the remaining worksheets and products, adding or removing data columns and applying the appropriate formats as necessary.

Using the POV as a Page Drop-down

When you have completed the grids, you can then set up the POV as a type of page drop-down to help users leverage the report.

- To set up the POV as a page drop-down:
- 1 In the POV on Sheet 1, select Options, then Select Members, then Market to display Member Selection.
- 2 Select all members from the Market dimension and click **OK**.
- 3 Dock the POV at the top of the Excel window.

You can now select members from the POV as needed and refresh the grid. As you do so, the data values on the grid are recalculated based on the selection in the POV:

In this example, California is about to be selected:

Tips

- Name the sheets to match the product line represented on each sheet.
- Use the Refresh All option to refresh all sheets in a workbook at one time, versus manually selecting and refreshing each sheet.

Note:

The techniques shown in Creating and Distributing Report Templates Using Free-Form, and in Answering Business Questions Using Free-Form work correctly whether you are connected to an Essbase or Financial Management or Hyperion Enterprise data source.

Retrieving Data into Asymmetric Reports

When you retrieve data into a worksheet, the resulting report can be either symmetric or asymmetric. Symmetric reports are characterized by repeating identical groups of members. For example, a symmetric report may contain Actual and Budget members nested below Year members (Qtr1, Qtr2, Qtr3, and Qtr4).

An asymmetric report is characterized by groups of nested members that differ by at least one member. There can be a difference in the number of members or in the names of members.

You can create asymmetric reports in one of the following ways:

Enter member names into the worksheet in free-form retrieval mode.

- Use a zoom in action with the "Within Selected Group" option selected from the Ad Hoc tab of the Options dialog box. See "Member Retention Options" on page 211 for information on setting this property.
- Suppress rows that contain missing values, zero values, or underscore characters during data retrievals. See "Suppressing Types of Data" on page 207 for information on suppressing rows.

Note that if you retrieve data into an asymmetric report, Smart View must perform additional internal processing to maintain the asymmetric layout. This processing may increase the retrieval time on large reports.



Accessing Dynamic Data Across **Microsoft Office**

In This Chapter

| Copying and Pasting Live Data into Word and PowerPoint | 159 |
|--|-----|
| Changing POV in Word and PowerPoint. | 160 |

Copying and Pasting Live Data into Word and PowerPoint

To present data, Word or PowerPoint is often preferable to Excel. Smart View enables you to copy data points from Excel and paste them into Word or PowerPoint, creating linked views (dynamic data) between applications.

Note:

When creating connections to data source providers, do not use semicolons (;) in connection names planned for using functions. An error occurs when pasting function data points, if connection names contain semicolons.

You can copy and paste data points from:

- Excel to Word and PowerPoint
- Word to Word and PowerPoint
- PowerPoint to Word and PowerPoint

The data points retain their original Excel-based query information, thus enabling users to perform data analysis. Word and PowerPoint can contain data points from multiple data sources, such as Essbase, Financial Management, Oracle BI EE, and Hyperion Enterprise within one document.

Notes:

- Dynamic data points are maintained only in Word and PowerPoint. If you copy and paste data points within Excel, the data points are not linked to the Excel grid.
- When copying and pasting from Word to PowerPoint, or vice versa, data is displayed in a straight line. The tabular format is preserved only when copying data from Excel into Word or PowerPoint.
- Excel formatting is preserved when data is pasted into Word and PowerPoint. Apply the formatting in Excel before copying and pasting data.

- You can use Word or PowerPoint tools to change number formatting.
- To copy and paste data from Excel to Word or PowerPoint:
- 1 In Excel, select a data cell or range (may or may not include members).
- 2 Click Copy Data or select Hyperion, then Copy Data Points.
- 3 Open a Word or PowerPoint document.
- 4 When asked if you want to create a connection, click Yes.
- 5 Click Paste Data or select Hyperion, then Paste Data Points.
- 6 Click Refresh or Refresh All, or select Hyperion, then Refresh or Refresh All.

The procedure for copying and pasting data points from Word to Word and PowerPoint or PowerPoint to Word and PowerPoint are similar.

Note:

When pasting data points into Word and saving the document in another format, such as a .htm or .mht file, you cannot refresh the data.

You can retrieve the Excel queries from which the data points were copied.

- To retrieve the Excel spreadsheets from which data points were copied:
- 1 In a Word or PowerPoint document into which Excel data points were pasted, select the data cells.
- 2 Click Visualize in Excel or select Hyperion, then Link View, then Visualize in Excel.
- 3 If asked to log on the data source, enter the username and password.

Excel displays the spreadsheet associated with the data cells. You can perform ad hoc analysis on the data.

You can view dynamic data in Visual Explorer.

Note:

Hyperion Enterprise does not support display of dynamic data in Visual Explorer.

- To display the data in Visual Explorer:
- 1 In a Word or PowerPoint document into which Excel data points were pasted, select the data cells.
- 2 Click Visualize in HVE or select Hyperion, then Link View, then Visualize in HVE...

Visual Explorer is displayed. See Visual Explorer Online Help.

Changing POV in Word and PowerPoint

After pasting Excel data points into Word or PowerPoint, you can use the POV Manager to change the POV.

- ➤ To change the POV in Word or PowerPoint:
- 1 In a Word or PowerPoint document into which Excel data points were pasted, click POV Manager or select Hyperion, then POV Manager.
- 2 In POV Manager, expand the POV list.
 - The data source connection information and active POV members are listed.
- 3 To change the POV, double-click a member in the POV list or click **Member Selector**.
 - The Member Selection dialog box is displayed.
- 4 Select a member using the Member Selection dialog box. See "Selecting Members for Ad Hoc Analysis" on page 97.
- 5 **Optional:** To modify the URL connection:
 - Under Connection, double-click the URL.
 - b. In Add Data Source, connect to a new data source.
- 6 Click Close.
- 7 Click Refresh or Refresh All, or select Hyperion, then Refresh or Refresh All.

12

Visual Explorer

In This Chapter

| Viewing Data in Graphical Format | 163 |
|--|-----|
| Viewing Data from Excel | 163 |
| Viewing Data from Word or PowerPoint | 164 |
| Starting Visual Explorer Independently of Smart View | 164 |
| Connecting to a Data Source from Visual Explorer | |

Viewing Data in Graphical Format

Visual Explorer provides Essbase users with powerful analytics in a highly graphical format. It enables users to unlock the information stored in multidimensional databases using a free-form canvas for exploring and analyzing data.

These graphical summaries answer questions about topics such as totals, comparisons, outliers, correlations, rankings, and trends—questions that you may have about the information in an Essbase database. You can choose to visualize data in various graphical formats including bars, lines, Gantt bars, shapes, colors, and tables.

Hyperion Visual Explorer is launched directly from the Hyperion menu in Smart View, using the "Visualize in HVE" command. By dragging interface elements, Essbase users can quickly summarize and visualize data from an Essbase database. Visual Explorer then enables you to pass data back to the Excel worksheet from which it was launched or to insert the data into a new Excel worksheet where additional analysis can be performed.

Note:

For detailed information on using Visual Explorer, refer to the online help that is accessed from the Help menu in Visual Explorer.

Viewing Data from Excel

- To view data in Excel using Visual Explorer:
- Open a worksheet in Excel.
- Connect to an Essbase data source.

- 3 Create a new report or use an existing one.
- 4 Select Hyperion, then Link View, and then Visualize in HVE.

The Visual Explorer interface is displayed.

5 In the Show Me! Alternatives dialog box, select a graphical format and click OK.

The data is displayed in the graphical format that you selected.

Viewing Data from Word or PowerPoint

Smart View enables you to copy data points from Excel to Word or PowerPoint and have those data points retain their original spreadsheet query information. See Chapter 11, "Accessing Dynamic Data Across Microsoft Office." You can also view the data in Word or PowerPoint in graphical format using Visual Explorer.

- To view data in Word or PowerPoint in Visual Explorer:
- 1 Select the data cells in Word or PowerPoint.
- 2 Select Hyperion, then Visualize in HVE.
- 3 In the **Show Me! Alternatives** dialog box, select a graphical format and click **OK**.

Visual Explorer displays the spreadsheet associated with the data cells in graphical format.

Starting Visual Explorer Independently of Smart View

You can start Visual Explorer as a stand-alone application, independent of Excel and Smart View.

- To start Visual Explorer as a stand-alone application:
- 1 Navigate to the location of your Smart View installation.

The default location is C:\Hyperion\SmartView.

2 In the bin folder, double-click hye, exe.

Follow the instructions in "Connecting to a Data Source from Visual Explorer" on page 164 to connect to an Essbase data source directly from Visual Explorer.

Connecting to a Data Source from Visual Explorer

If you are not logged into an Essbase data source, you can log in from Visual Explorer.

- To connect to a data source from Visual Explorer:
- 1 From Oracle Essbase Visual Explorer, select **Data** then **Connect to Data Source**.
- 2 Follow the procedure in Chapter 4, "Connecting to Data Sources" to complete the connection.

Importing Reporting and Analysis Documents into **Microsoft Office**

In This Chapter

| Importing Reporting and Analysis Documents | 165 |
|--|-----|
| Editing and Refreshing Documents. | 166 |
| Financial Reporting and Web Analysis Import Formats | 167 |
| Adding Security Certificates for SSL-enabled EPM Workspace Servers | 168 |
| Importing Interactive Reporting Documents | 168 |
| Importing Financial Reporting Documents | 172 |
| Importing Production Reporting Documents | 176 |
| Importing Web Analysis Documents | 179 |
| Using Smart Tags to Import Reporting and Analysis Documents | 181 |

Importing Reporting and Analysis Documents

Using Smart View, you can import Reporting and Analysis documents into Microsoft Excel, Word, or PowerPoint.

- For Financial Reporting and Web Analysis, you can import reports.
- For Interactive Reporting, you can import charts, dashboards, and reports. Using the latest run of BQY jobs, Interactive Reporting supports refresh capabilities.
- For Production Reporting, you can import jobs and job outputs.

Important tasks:

- "Importing Interactive Reporting Documents" on page 168
- "Importing Financial Reporting Documents" on page 172
- "Importing Production Reporting Documents" on page 176
- "Importing Web Analysis Documents" on page 179
- "Using Smart Tags to Import Reporting and Analysis Documents" on page 181

Editing and Refreshing Documents

In Office, the Hyperion menu (Hyperion ribbon in Office 2007) provides the ability to edit and refresh documents that were previously imported from EPM Workspace. The Hyperion menu or ribbon provides the following edit and refresh options:

- Edit—change filters, POVs, or parameters of embedded EPM Workspace documents.
- Refresh—refresh the selected job with the latest EPM Workspace data. Only the job selected in Office is updated; not the entire Office document.
- Refresh All—update all jobs in the Office document.

General edit and refresh behavior

- After the initial import into Office, you can delete any number of pages. Then, when you perform a Refresh, an update is performed on the remaining pages; the deleted pages are not reinstated into the document. For example, if you delete page 2 of 4 pages, only pages 1, 3, and 4 are updated.
- If during Edit or refresh the number of imported pages are less than the original import, the removed pages display as blanks pages in Office.
- If an Edit or Refresh results in additional Workspace pages, those pages are appended to the in Office.
- With refresh, formatted pages such as headings and comments are retained in Word and PowerPoint. In Excel, formatting is not retained on refreshed pages.

Maintaining cell references during document refresh

In Excel, a customized worksheet that references imported document cells or ranges is updated when you execute a Refresh All on the imported documents. For example, imported worksheet A and B are referenced in customized worksheet C. When you execute Refresh All on worksheet A and B, worksheet C is refreshed with updated data from worksheets A and B.

Refreshing Reporting and Analysis Documents

Refreshing updates the report with the latest data from EPM Workspace.

Refresh behavior in Production Reporting and Interactive Reporting:

- In Word, if a report is selected, the entire report is refreshed. If no report is selected, the first report found in the document is refreshed. The first report is not necessarily the report at the beginning of the document.
- In PowerPoint, if no report is selected, the first report found in the slide is updated.

When refreshing job outputs in Production Reporting, new outputs in EPM Workspace are updated.

Refresh behaviors in Financial Reporting and Web Analysis:

You must select a page in the report to refresh. In Word and PowerPoint, if you do not select any pages when refreshing, a message is displaying stating that no pages are updated.

- If you select Refresh, all pages of the report are refreshed. If you select Refresh All, then all reports in the document are refreshed.
- To refresh EPM Workspace documents in Excel, Word, or PowerPoint, perform an action:
- 1 Select Hyperion, then Refresh to update the selected Reporting and Analysis document, including all pages associated with that document. For Office 2007, click Refresh in the Review section of the Hyperion ribbon.
- 2 Or, select Hyperion, then Refresh All to update all Reporting and Analysis documents. For Office 2007, click Refresh All in the Review section of the Hyperion ribbon.

Refreshing Reporting and Analysis documents against Essbase or Financial Management connections

Note:

This applies to Financial Reporting and Web Analysis reports imported into query-ready HTML.

Refreshing against Essbase or Financial Management connections updates the report with the latest data from Analytic Services and enables you to perform ad hoc analysis on the Reporting and Analysis document, such as retrieving, zooming, or pivoting data. See Chapter 9, "Working with Ad Hoc Analysis".

Refresh for a report imported in query-ready HTML applies to the current page and not all pages.

Important Tasks

- "Editing Interactive Reporting Documents" on page 171
- "Editing Financial Reporting Documents" on page 174
- "Editing Production Reporting Jobs" on page 178
- "Editing Web Analysis Documents" on page 180

Financial Reporting and Web Analysis Import Formats

You can import Financial Reporting and Web Analysis documents as fully-formatted HTML, which you can display in Excel, or in query-ready HTML, which enables you to connect to Financial Management or Essbase data sources and run queries.

When you import Reporting and Analysis documents as query-ready HTML, the selected pages of the current data object is converted to HTML, and Hyperion-specific formatting is removed. Thus, Smart View can re-query the data source independent of the Web application.

When you import Reporting and Analysis documents as fully formatted HTML, the selected pages of the current data object is converted to HTML, and Hyperion formatting definitions and calculated members are retained. Thus, Smart View cannot directly query the data source, but Hyperion content can be leveraged by Microsoft Office applications.

Tip:

After importing an image in Word or PowerPoint, use the Microsoft Office Format Picture option to format it; for example, to crop and resize. The Format Picture settings are preserved, even after you refresh the image.

Adding Security Certificates for SSL-enabled EPM Workspace Servers

When accessing EPM Workspace servers running on HTTPS protocols, the server's security certificate must be issued by a trusted company. To avoid unnecessary prompts when importing documents, you must add the security certificate into the Internet Explorer certificate path on the Smart View client's machine.

- To add security certificates:
- 1 In Internet Explorer, connect to EPM Workspace; for example:

https://<workspace_webserver>.hyperion.com/workspace/index.jsp.

- 2 If prompted with the warning message, "The security certificate was issued by a company you have not chosen to trust," select View Certificate.
- 3 Select Certification Path.
- 4 Ensure that all certificates in the certification path have been trusted.

Importing Interactive Reporting Documents

- "Importing Interactive Reporting Documents into Excel" on page 169
- "Importing Interactive Reporting Documents into Word and PowerPoint" on page 170
- "Editing Interactive Reporting Documents" on page 171

Imported Interactive Reporting documents are section-specific.

Table 7 Interactive Reporting Import Object Types

| Section | Excel | Word, PowerPoint |
|-----------|----------------|------------------|
| Table | Formatted data | NA |
| Results | Formatted data | NA |
| Chart | Formatted data | Image |
| Pivot | Formatted data | NA |
| Report | Formatted data | Image |
| Dashboard | Image | Image |

| Section | Excel | Word, PowerPoint |
|------------|---|------------------|
| Query | NA | NA |
| CubeQuery | Query ready (Internet Explorer only, not supported by Firefox) Formatted data | N/A |
| Data model | NA | NA |

The following restrictions apply when Interactive Reporting documents are imported into Excel:

- Hidden sections are displayed during import.
- Importing dashboard sections into Excel resizes A1 cells.
- Importing report sections into Excel places chart images before tables
- Importing into Excel may not preserve colors correctly.
- Results sections that contain the euro currency format do not import into Excel.
- Results sections with "+" in their name do not import.

Importing Interactive Reporting Documents into Excel

- To import Interactive Reporting documents into Excel:
- 1 Connect to a EPM Workspace data source.

See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.

- 2 If you have multiple connections, select Hyperion, then Active Connections, and select a data source. For Office 2007, click Activate in the Connections section of the Hyperion ribbon.
- 3 Select Hyperion, then Reporting and Analysis Document, then Import. For Office 2007, click R&A Import in the Task section of the Hyperion ribbon

The Import Workspace Document dialog box is displayed.

4 In Select a Document, expand the repository, select an Interactive Reporting document, and click OK. A wizard screen is displayed.

Note:

Some wizard screens do not apply to some documents.

- 5 In Selections, select the section for importing.
- In **Actions**, select an option:
 - Refresh and Preview, to change filters or values prior to previewing the document
 - Preview, to preview the document with default settings If you are importing a CubeQuery section in query ready format, do not select this options.

- 7 Click Next.
- 8 If you select Refresh and Preview, steps 9-10 are required. If you select Preview, skip to step 11
- 9 If user authentication is required to change filters, such as variable, value, or option in the document's settings, in Specify Database Credentials, enter the username and password.password, and select Next.

The connection name is displayed in parentheses (for example, Sample.oce).

- 10 In Specify Filter, select a value and click Next.
- 11 To import all pages of the document, leave the All Pages field check enabled.
- 12 If your document contains multiple pages, select Split pages across worksheets to display each page on a separate Excel worksheet.
- 13 In the Import section As drop-down, select the method for importing: Fully Formatted or Query Ready.
 - When you export content as query-ready HTML, the current page of the current CubeQuery section is converted to HTML and Hyperion-specific formatting is removed. This enables to requery the data source independent of the Web application.
 - When you export content as Formatted HTML, the current page of the CubeQuery section is converted to HTML with the Hyperion formatting definitions and calculated members. This formatting content prevents Smart View from directly querying the data source, but enables Hyperion content to be leveraged by Office applications.

The query ready option is only available for a CubeQuery section for a Refresh and Preview action.

14 Click Finish.

The document is displayed in Excel.

Importing Interactive Reporting Documents into Word and PowerPoint

- To import Interactive Reporting documents into Word:
- 1 Connect to a EPM Workspace data source.

See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.

2 Select Hyperion, then Reporting and Analysis Document, then Import. For Office 2007, click R&A Import in the Task section of the Hyperion ribbon.

The Import Workspace Document dialog box is displayed.

3 In Select a Document, expand the repository, select an Oracle's Hyperion® Interactive Reporting document, and click OK.

Note:

Some wizard screens do not apply to some documents.

- In Select an Action, select an option:
 - Process and Preview, to change filters or values prior to previewing the document
 - Preview, to preview the document with default settings
- Click Next.
- 6 If you selected Process and Preview:
 - If user authentication is required to change filters, such as variable, value, or option in the document settings, in Specify Database Credentials, enter the username and password, and click Next.

The connection name is displayed in parentheses (for example, Sample.oce).

- b. In Specify Filters, select a value.
- 7 Click Apply, and click Next.
- 8 In Preview, to import a page, select a page from the drop-down list located in the upper left of the data object.:
- 9 Optional: To import all pages of the document, select All Pages.
- 10 Click Finish.

The document is imported.

Editing Interactive Reporting Documents

- To edit Interactive Reporting documents in Excel, Word, and PowerPoint:
- 1 Select a page.

Note:

When editing a report, select a page from the report, then Edit. If you do not select a page, a message is displayed stating that no pages are updated.

- 2 Select Hyperion, then Reporting and Analysis Document, then Edit. For Office 2007, click R&A Edit in the Task section of the Hyperion ribbon.
- 3 If you selected Process and Preview:
 - If user authentication is required to change filters, such as variable, value, or option in the document settings, in Specify Database Credentials, enter the username and password, and click Next.

The connection name is displayed in parentheses (for example, Sample.oce).

- In Specify Filters, select a value. b.
- 4 Click Apply, and click Next.
- 5 In Preview, to import a page, select a page from the drop-down list located in the upper left of the data object.
- 6 Click Finish.

Importing Financial Reporting Documents

- "Financial Reporting and Web Analysis Import Formats" on page 167.
- "Importing Financial Reporting Documents into Excel" on page 172
- "Importing Financial Reporting Documents into Word and PowerPoint" on page 173
- "Editing Financial Reporting Documents" on page 174

Table 8 **Financial Reporting Import Document Types**

| Document Type | Excel | Word, PowerPoint |
|-------------------------|------------------------------|------------------|
| Report | Fully formatted, query-ready | Image |
| Snapshot report | Fully formatted | Image |
| Book | NA | NA |
| Snapshot Book | NA | NA |
| Batch | NA | NA |
| Grid Object | NA | NA |
| Image Object | NA | NA |
| Chart Object | NA | NA |
| Text Object | NA | NA |
| Row and Column template | NA | NA |

Importing Financial Reporting Documents into Excel

- To import Financial Reporting documents into Excel:
- 1 Connect to an EPM Workspace data source.

See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.

- 2 If you have multiple connections, select Hyperion, then Active Connections and select a data source to activate. For Office 2007, click Activate in the Connections section of the Hyperion ribbon.
- 3 Select Hyperion, then Reporting and Analysis Document, then Import. For Office 2007, click R&A Import in the Task section of the Hyperion ribbon.

The Import Workspace Document dialog box is displayed.

4 In Select a Document, expand the repository, select a Financial Reporting document, and click OK. The document is previewed in the Import Workspace Document window.

Note:

Some options may not be available for some documents.

5 If the Preview User Point of View is displayed, preview the current POV or change the members of the POV.

Note:

To display this screen, select Preview in EPM Workspace preferences, for User Point of View.

- 6 Click Next.
- 7 Optional: If you want to change the default value, in **Respond to Prompts**, make a selection for prompts, and click Next.

Note:

This screen is displayed only if the document contains prompts.

- 8 Optional: In Preview from Grid POV, change the POV by selecting a POV.
- 9 Change the page dimension by selecting Page.
- 10 To import all pages of the document, select All Pages to import all pages of the document.
- 11 To display each page on a separate Excel worksheet, select Split Pages across worksheets.
- 12 In **Import Document As**, select an option:
 - Fully-Formatted (displays reports in a fully-formatted HTML)
 - Query-Ready (enables you to run ad hoc analysis on reports when connected to Financial Management and Essbase data sources)
- 13 Click Finish.

The document is imported into Excel. If you used the Fully-Formatted option, you can only view the Reporting and Analysis document. If you used the Query-Ready option, then connect to a Financial Management or Essbase data source, you can perform ad hoc analysis, such as retrieving, zooming, or pivoting data. See Chapter 9, "Working with Ad Hoc Analysis".

Importing Financial Reporting Documents into Word and PowerPoint

- To import Financial Reporting documents into Word and PowerPoint:
- 1 Connect to an EPM Workspace data source.

See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.

2 Select Hyperion, then Reporting and Analysis Document, then Import. For Office 2007, click R&A Import in the Task section of the Hyperion ribbon.

The Import Workspace Document dialog box is displayed.

3 In Select a Document, expand the repository, select a Oracle Hyperion Financial Reporting, Fusion Edition document, then click OK.

The document is previewed in the Import Workspace Document window.

Note:

Some screens do not apply to some documents.

4 If the Preview User Point of View screen is displayed, preview the current POV or change the members of the POV by selecting a member.

Note:

To display this screen, select Preview in EPM Workspace preferences, for User Point of View.

5 Optional: If you want to change the default value, in **Respond to Prompts**, make a selection for prompts, and click Next.

Note:

This screen is displayed only if the document contains prompts.

- 6 In **Preview** from **Grid POV**, change the POV by selecting a POV.
- 7 Change the page dimension by selecting **Page**.
- 8 Select All Pages to import all pages of the document.
- 9 In Import Document As, select Image to import the document as an image.
- 10 Click Finish.

The document is imported.

Editing Financial Reporting Documents

- To edit Financial Reporting documents in Excel, Word, and PowerPoint:
- 1 Select a page.

Note:

When editing a report, select a page from the report, then Edit.

- 2 Select Hyperion, then Reporting and Analysis Document, then Edit. For Office 2007, click R&A Edit in the Task section of the Hyperion ribbon.
- 3 If the Preview User Point of View screen is displayed, preview the current POV or change the members of the POV.

Note:

To display this screen, select Preview in EPM Workspace preferences, for User Point of View.

Optional: If you want to change the default value, in Respond to Prompts, make a selection for prompts, and click Next.

Note:

This screen is displayed only if the document contains prompts.

- 5 If you want to change the POV, in **Preview** from **Grid POV** select a POV.
- 6 Click Finish.

Creating Templates in PowerPoint Documents

You can create PowerPoint template documents by importing one or more Financial Reporting reports to the presentation. Every Create Template action creates a new PowerPoint slide with a report name to show where it will be placed when Refresh Template is used. The template PowerPoint presentation can be saved for future use.

- To create a template:
- 1 Open PowerPoint.
- 2 Connect to a Reporting and Analysis provider using Hyperion, then Data Source Manager.
- 3 Select Hyperion, then Reporting and Analysis Document, then Create Template.
- 4 In Import Workspace Document, select a Financial Reporting document.
 - Optional: To import all pages of the document, select All Pages. A separate slide is created for each page.
 - To import the current screen presentation, clear All Pages.
- 5 Optional: To use the Workspace point of view, select Refresh Using Workspace Point of View.
- 6 Click **OK**. The document name is imported into the PowerPoint presentation.

Refreshing PowerPoint Templates

When you refresh a template, the template is replaced with a new presentation and each Financial Reporting report placeholder in the template is replaced with the current Financial Reporting report using the current Workspace Point of View.

- To refresh a template:
- 1 Open the PowerPoint presentation containing the template.
- 2 Connect to a Reporting and Analysis provider.
- 3 Select Hyperion, then Reporting and Analysis Document, then Refresh Template.
- Edit and save the PowerPoint presentation as needed.

Importing Production Reporting Documents

Production Reporting documents consist of jobs and job outputs, which you can import into Excel, Word, and PowerPoint.

- "Importing Production Reporting Jobs into Excel" on page 176
- "Importing Production Reporting Jobs into Word and PowerPoint" on page 177
- "Importing Production Reporting Job Outputs into Excel, Word, and PowerPoint" on page 178
- "Editing Production Reporting Jobs" on page 178

Table 9 **Production Reporting Import Object Types**

| Object Type | Excel | Word, PowerPoint |
|-------------|----------------|------------------|
| Job | Formatted data | Image |
| Job output | Formatted data | Image |

Some limitations exist for importing:

- Images and charts are not imported into Excel.
- Secure jobs are supported, but jobs imported as generic jobs are not supported.

Importing Production Reporting Jobs into Excel

- To import Production Reporting jobs into Excel:
- 1 Connect to an EPM Workspace data source. See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.
- 2 If you have multiple connections, select Hyperion, then Active Connections and select a data source to activate. For Office 2007, click Activate in the Connections section of the Hyperion ribbon.
- 3 Select Hyperion, then Reporting and Analysis Document, then Import. For Office 2007, click R&A Import in the Task section of the Hyperion ribbon.

The Import Workspace Document dialog box is displayed.

4 In Select a Document, expand the repository, select a Production Reporting job, then click OK.

The import wizard screen is displayed.

Note:

Depending on the document, some screens may not be applicable.

5 If the Specify Parameters screen is displayed, define the job parameters, and click Next.

Note:

This screen is displayed only if the job contains parameters.

- 6 In Preview, to import a page, select a page from the drop-down list located in the upper left of the data object.
- 7 To import all pages of the job, select All Pages.
- Select Split Pages across worksheets to display each page on a separate Excel worksheet.
- 9 Click Finish.

The document is displayed in Excel.

Importing Production Reporting Jobs into Word and PowerPoint

The procedures for importing Production Reporting jobs into Word and PowerPoint are similar.

- To import Production Reporting jobs into Word and PowerPoint:
- 1 Connect to an EPM Workspace data source.

See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.

2 Select Hyperion, then Reporting and Analysis Document, then Import. For Office 2007, click R&A Import in the Task section of the Hyperion ribbon.

The Import Workspace Document dialog box is displayed.

3 In Select a Document, expand the repository, select a Reporting and Analysis document, then click OK. The import wizard is displayed.

Note:

Some screens may not apply to some documents.

4 If the Specify Parameters screen is displayed, define the job parameters, and click Next.

Note:

This screen is displayed only if the job contains parameters.

- 5 In Preview, to import a page, select a page from the drop-down list located in the upper left of the data object.
- 6 To import all pages of the job, select All Pages.

For Word, Split pages across pages is disabled. For PowerPoint, Split pages across slides is selected and disabled because by default, the pages from jobs or job outputs always split across pages and slides.

7 Click Finish.

The job is imported.

Importing Production Reporting Job Outputs into Excel, Word, and PowerPoint

- To import Production Reporting job outputs into Excel, Word, and PowerPoint:
- 1 Connect to a EPM Workspace data source. See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.
- 2 If you have multiple connections, select Hyperion, then Active Connections and select a data source to activate. For Office 2007, click Activate in the Connections section of the Hyperion ribbon.
- Select Hyperion, then Reporting and Analysis Document, then Import. For Office 2007, click R&A Import in the Task section of the Hyperion ribbon.

The Import Workspace Document dialog box is displayed.

In Select a Document, expand the repository, select a Production Reporting job output, then click OK. The job output is imported.

Editing Production Reporting Jobs

You can edit imported Oracle's Hyperion® SQR® Production Reporting jobs, but not job outputs. You can edit only job parameters.

- To edit Production Reporting jobs:
- 1 Select a page.

Note:

When editing a report, select a page from the report, then Edit. If you do not select a page, a message is displayed stating that no pages are updated.

2 Select Hyperion, then Reporting and Analysis Document, then Edit. For Office 2007, click R&A Edit in the Task section of the Hyperion ribbon.

The Import Workspace Document dialog box is displayed.

If the Specify Parameters screen is displayed, define the job parameters, and click Next.

Note:

This screen is displayed only if the job contains parameters.

In Preview. view the job.

If you deleted any imported pages, edit updates only the remaining pages of the job.

Click Finish.

The job is updated.

Importing Web Analysis Documents

Web Analysis includes five data object display types, but Smart View can import only three (spreadsheet, chart, and pinboard). Smart View cannot import free form grid and SQL spreadsheets. See "Financial Reporting and Web Analysis Import Formats" on page 167.

- "Importing a Web Analysis Document or Document Objects" on page 179
- "Editing Web Analysis Documents" on page 180

Table 10 **Web Analysis Import Document Type**

| Document Type | Excel | Word, PowerPoint |
|---------------|------------------------------|------------------|
| Report | Fully formatted, query-ready | Image |

Table 11 Web Analysis Import Data Object Type

| Data Object | Excel | Word, PowerPoint |
|-------------|-------------------|------------------|
| Spreadsheet | Data + formatting | Image |
| Chart | Data + formatting | Image |
| Pinboard | Data + formatting | Image |

Importing a Web Analysis Document or Document Objects

Using Smart View in Excel, you can import one or all document pages or multiple data objects with one or more pages from a Web Analysis document residing in the Workspace repository. All Web Analysisdata objects (spreadsheet, chart, pinboard) are imported as Excel spreadsheets. Freeform Grid and SQL spreadsheets cannot be imported.

- To import Web Analysis data objects:
- 1 Connect to a EPM Workspace data source. See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.
- 2 If you have multiple connections, select Hyperion, then Active Connections and select a data source to activate. For Office 2007, click Activate in the Connections section of the Hyperion ribbon.
- 3 Select Hyperion, then Reporting and Analysis Document, then Import. For Office 2007, click R&A Import in the Task section of the Hyperion ribbon.
 - The **Import Workspace Document** dialog box is displayed.
- 4 In Select a Document, expand the repository, select a Web Analysis document, and click OK.
- 5 If database credentials are not saved with the Web Analysis document, then the Specify Database Credentials page is displayed where you are required to enter valid log on credentials to data sources used in the report. If a report has only one data source and you skip entering credentials, the report is not imported. If you have data objects with different data sources in one report and only want to import one of the data objects, you can enter the credentials for the data objects you want to import and skip credential for the

data object you do not wish to import. Enter the user name and password or select Skip to skip entering credentials to any of the data sources, and click Next.

Tip:

Select Save Credentials to save credentials with a Web Analysis document. It enables you to refresh an imported document later. Currently, you cannot refresh imported documents without saving credentials.

- 6 In Preview, when selecting objects to import for Microsoft Excel, Word, and Powerpoint:
 - Select individual data objects, by clicking the check box located in the top left corner of each report object OR select all data objects by clicking the All Objects check box.
 - Select Split Objects across worksheets to create a new worksheet for each report object OR deselect Split Objects across worksheets to placed all report objects in the same worksheet.
 - Select a page to import from the drop-down list located in the top of each selected to import data objects OR select All Pages to import all pages of all selected to import data objects.
 - Select Split Pages across Worksheets to create a new worksheet for each import page OR deselect Split Pages across Worksheets to place all imported pages of each data object in the same worksheet.
- 7 In Preview, when selecting object to import for Microsoft Word and PowerPoint, select Import Screen to import a screen print of the entire report.
- 8 For Microsoft Excel, In **Import Document As**, select an option:
 - Fully Formatted (imports reports in fully-formatted HTML). You can connect to EPM System at any time and refresh the imported document for current data.
 - Query-Ready (imports reports in query-ready HTML). You can connect to Financial Management or Essbasedata source to get data directly and perform ad hoc analysis, such as retrieving, zooming, and pivoting data.
- 9 Click Finish. The document is imported. You can then connect to Oracle Hyperion Enterprise Performance Management System at any time and refresh the imported document with current data.

Editing Web Analysis Documents

- To edit Web Analysis documents:
- 1 Select a page (Excel) or an image (Word or PowerPoint).

Note:

When editing a report, select a page (Excel) or an image (Word or PowerPoint) from the report, then Edit. If you do not select a page or an image, a message is displayed stating that no pages are updated.

2 Select Hyperion, then Reporting and Analysis Document, then Edit. For Office 2007, click R&A Edit in the Task section of the Hyperion ribbon.

3 If database credentials are not saved with the Web Analysis document, then the Specify Database Credentials page is displayed. In Specify Database Credentials, enter the user name and password, or select Skip, then click Next.

Tip:

You can select Save Credentials to save them with the Oracle's Hyperion® Web Analysis document.

- 4 Select a data object (spreadsheet, chart, or pinboard) to import.
- 5 In Preview, to import a page, select a page from the drop-down list located in the upper left of the data
- 6 Select All Pages to import all pages of the document. Leave the box cleared to import only the current page.
- 7 Select Split Pages across worksheets to display each page on a separate worksheet (Excel only).
- 8 In Import Document As, select:
 - Fully Formatted (Excel only)
 - Query-Ready (Excel only)
 - Image (Word and PowerPoint)
- Click Finish.

Using Smart Tags to Import Reporting and Analysis Documents

For Microsoft Office 2003, smart tags provide an alternative way of importing Reporting and Analysis documents, enabling you to import fully-formatted image objects from EPM Workspace into documents.

Smart tags are supported in Outlook 2003 only if the e-mail editor uses Microsoft Word. By default, smart tags are turned off. Smart tags are not supported in Office versions prior to Microsoft Office 2003.

- "Using Smart Tags to Import Reporting and Analysis Documents" on page 181
- "Removing Smart Tags for a Single Instance" on page 195
- "Stopping Hyperion Smart Tag Recognition" on page 195
- "Recognizing Smart View Smart Tags Again" on page 196
- To turn on smart tags:
- 1 Select Tools, then AutoCorrect Options.
- 2 Select Smart Tags.
- 3 Select Label text with smart tags.
- Click OK.

- To import Reporting and Analysis documents using smart tags:
- 1 Open a Microsoft Office document.
- 2 Select Hyperion, then Data Source Manager and connect to a Oracle Enterprise Performance Management Workspace, Fusion Edition data source. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.
 - See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.
- 3 Type smartview anywhere in the document, then move the mouse over the word.
 - The smart tags action icon is displayed. The smart view keyword is not case-sensitive.
- 4 Click the smart tags action icon to display the **Smart View** menu.
- 5 From Smart View, select Reporting and Analysis Content.
 - The Import Workspace Document dialog box is displayed.
- 6 Follow the procedures for importing Reporting and Analysis documents after opening the Import Workspace **Document** dialog box:
 - "Importing Interactive Reporting Documents" on page 168
 - "Importing Financial Reporting Documents" on page 172
 - "Importing Production Reporting Documents" on page 176
 - "Importing Web Analysis Documents" on page 179

14

Functions

In This Chapter

| About Functions | 184 |
|--------------------------------------|-----|
| The Function Builder | 18 |
| HsGetValue | 189 |
| HsSetValue | 189 |
| HsCurrency | 190 |
| HsDescription | 190 |
| HsLabel | |
| HsGetText | 193 |
| HsSetText | 19 |
| Accessing Functions Using Smart Tags | 192 |
| Creating Functions Manually | |
| Editing Functions | 19 |
| Running Functions | 19 |
| Common Function Error Codes | 198 |

You can use functions to retrieve data from a Microsoft Excel worksheet and save it to your data source, or send data from a data source to a worksheet. You can also use functions to retrieve currency values, display member descriptions and labels, and send and retrieve cell text. Functions are available when accessing Financial Management, Essbase, and Hyperion Enterprise data source providers. You can also use smart tags to access functions.

Note:

When creating connections to data source providers, do not use semicolons (;) in connection names planned for using functions. An error occurs when using Function Builder to create functions.

When a worksheet that contains saved functions is opened on a different machine than the one on which it was created, the functions include the path of the original machine. Smart View automatically updates the function paths when you open the worksheet.

Note:

If you are prompted to manually update the links, click Cancel and the links update automatically. Functions are not updated automatically if the Excel "Ask to Update Automatic Links" option is selected. In that case, you must manually update functions using the Excel Links option. Functions on a protected worksheet are not automatically updated.

About Functions

You create functions by either of the following:

- The Function Builder (see "The Function Builder" on page 185).
- Creating function syntax manually (see "Creating Functions Manually" on page 196).

When you use the Function Builder to create functions, you can validate the function syntax before you paste it to a worksheet. When you create functions manually in worksheet cells, the functions are validated only when you refresh them.

HsGetValue: Retrieves data from a data source for a given Point of View. Supported for Financial Management, Hyperion Enterprise, and Essbase

HsSetValue: Sends values to the data source for a given Point of View. Supported for Financial Management, Hyperion Enterprise, and Essbase

HsCurrency: Retrieves the entity currency for the selected members. Supported for Financial Management and Hyperion Enterprise.

HsDescription: Displays the description for the default Point of View member. Supported for Financial Management and Hyperion Enterprise.

HsLabel: Displays the label for the default Point of View member. Supported for Financial Management and Hyperion Enterprise.

HsGetText: Retrieves cell text from the data source. Supported for Financial Management.

HsSetText: Sends cell text to the data source. Supported for Financial Management.

For each function, you can specify a connection and a full or partial Point of View. If you do not specify a connection, the system uses the default connection. Some functions also require a value. For example, the HsSetValue function requires that you set a value in addition to the connection and Point of View.

If you do not specify a dimension in a function, the system uses the dimension member specified in the POV Manager. See Chapter 17, "Using the POV Manager".

If specified, the connection must precede the Point of View. The Point of View is made up of dimension#member pairs, where the pound symbol (#) is a separator between dimension and member, for example, Entity#Connecticut. Parent-child relationships can be represented with a period, for example, Entity#UnitedStates.Maine.

The connection and Point of View can be grouped as one parameter, for example "My_connection;Entity#UnitedStates". Alternatively, they can be split up into several function parameters, for example, "My_connection", "Entity#UnitedStates", "Account#Sales". If the connection and Point of View are in the same function parameter, the connection and each

dimension#member pair are separated by a semi-colon (;) character, for example, "My_connection;Entity#UnitedStates;Account#Sales".

Examples

```
HsGetValue(B4, "HFM01", "Scenario #Actual; Year #2004; Period #July; View #YTD;
Entity#UnitedStates.Connecticut; Value#USD; Account#Sales; ICP# [ICP
None];Custom1#GolfBalls;Custom2#Customer2;Custom3#[None];Custom4#
Increases")
HsGetValue(B4, "HE01", "Scenario#Actual; Period#July; View#YTD;
Entity#UnitedStates.Connecticut;Account#Sales")
HsCurrency("HFM03","Entity#UnitedStates.Connecticut","Value#USD")
HsSetText
("text", "Scenario #Budget; Year #2004; Period #July; View #YTD; Entity #United States
.Connecticut; Value#USD; Account#Purchases; ICP#[ICP
None];Custom1#GolfBalls;Custom2#Customer2;Custom3#[None];Custom4#
Increases")
```

You can also use cell references in functions. For example, the following HsSetValue function uses cell references for the data value in the H4 cell and for the period:

```
HsSetValue (H4, "Comma"; "Period#"&B$2&"; Entity#UnitedStates. Connecticut")
```

The Function Builder

The Function Builder provides a graphical interface for creating functions. From the Function Builder, you can select a function, then select dimension members for the Point of View. You can view and validate the function before you paste it to the selected worksheet.

If a dimension does not apply for a specific function, the dimension is not available for selection. If you do not select a member for a dimension, the system uses the default members from the Point of View.

For information on selecting members, see "Selecting Members for Functions" on page 188.

You can use cell references for all functions except HsLabel. When you enter a cell number, for example, B3, in the Member text box and select Cell Reference as the Member Type, the function uses the value from the referenced cell.

Note:

Before you begin this procedure, connect to all data sources that your functions will access.

- To create functions using the Function Builder:
- 1 Select the spreadsheet cell in which you want to paste the function.
- 2 Select Hyperion, then Functions, then Function Builder. For Office 2007, click Function Builder in the Task section of the Hyperion ribbon.

3 From **Select Connection**, select a data source.

Only connected data sources are displayed.

- 4 From the list of functions, select one of the following functions and then proceed to step 5:
 - To retrieve values, select HsGetValue.
 - To send values, select HsSetValue.
 - To retrieve the entity currency, select HsCurrency.
 - To display the description for the default Point of View member, select HsDescription.
 - To display the label for the default Point of View member, select HsLabel.
 - To retrieve cell text, select HsGetText.
 - To submit cell text, select HsSetText.

Note:

When connected to Provider Services, only HsGetValue and HsSetValue are displayed in Function Builder.

5 Depending on the function you selected in step 4, select parameters as follows:

| Function | Parameter Selection |
|------------|---|
| HsGetValue | a. Select a dimension. |
| | b. In the Member column, do one of the following: |
| | Double-click to enter a member name. |
| | Click Select Member to select members as described in "Selecting Members for Functions" on page 188. |
| | To use a cell reference, enter a cell identity. |
| | c. Double-click in the Member Type column. From the drop-down list, select Member or Cell Reference. |
| HsSetValue | a. Select a dimension. |
| | b. In the Member column, do one of the following: |
| | Double-click to enter a member name. |
| | Click Select Member to select members as described in "Selecting Members for Functions" on page 188. |
| | To use a cell reference, enter a cell identity. |
| | c. Double-click in the Member Type column, then from the drop-down list, select either Member or Cell Reference. |
| | d. Select Data or Cell Reference and enter the value to send. |
| HsCurrency | a. Select the Entity or Value dimension. |
| | b. In the Member column, do one of the following: |
| | Double-click to enter a member name. |
| | Click Select Member to select members as described in "Selecting Members for Functions" on page 188. |

| Function | Parameter Selection | | |
|---------------|--|--|--|
| | To use a cell reference, enter a cell identity. | | |
| | Double-click in the Member Type column. Fom the drop-down list, select Member or Cell Reference. | | |
| HsDescription | a. From the Dimension drop-down list, select a dimension. | | |
| | b. In the Member column, do one of the following: | | |
| | Double-click to enter a member name. | | |
| | Click Select Member to select members as described in "Selecting Members for Functions" on page 188. | | |
| | To use a cell reference, enter a cell identity. | | |
| | Double-click in the Member Type column. From the drop-down list, select Member or Cell Reference. | | |
| HsLabel | From the Dimension drop-down list, select a dimension. | | |
| HsGetText | a. Select a dimension. | | |
| | b. In the Member column, do one of the following: | | |
| | Double-click to enter a member name. | | |
| | Click Select Member to select members as described in "Selecting Members for Functions" on page 188. | | |
| | To use a cell reference, enter a cell identity. | | |
| | Double-click in the Member Type column. From the drop-down list, select Member or Cell Reference. | | |
| HsSetText | a. Select a dimension. | | |
| | b. In the Member column, do one of the following: | | |
| | Double-click to enter a member name. | | |
| | Click Select Member to select members as described in "Selecting Members for Functions" on page 188. | | |
| | To use a cell reference, enter a cell identity. | | |
| | Double-click in the Member Type column. From the drop-down list, select Member or Cell Reference. | | |
| | d. Select Comment or Cell Reference and enter the text to send. | | |

- 6 Click Add to Function.
- 7 Optional: To add another function, click Add to Function.
- 8 Optional: If you have manually edited the function in the Function text box, to validate the function syntax before inserting it into the worksheet, click Validate Syntax.

Note:

Validate Syntax validates only the syntax you are using for the function. It does not validate the members you have selected.

9 Click **OK** to paste the function into the worksheet.

10 To execute the function, select Hyperion, then Refresh or Refresh All. For Office 2007, click Refresh or Refresh All in the Review section of the Hyperion ribbon.

Note:

Selecting Refresh updates the sheet, but removes the function.

Selecting Members for Functions

You use the Member Selection dialog box to select dimension members. You can filter the list of members and view labels or descriptions for dimension members. For example, when you select to view descriptions, the P_Series member is displayed as "P_Series - Phones and PDAs".

You can use the Select button to select multiple members simultaneously. For example, you can select the Children option for all immediate dependents of a member, select the Base Members option for all base members, or select both the Children and Base Members options to select all dependents.

You can also search for dimension members. See step 10 on page 189.

- To select dimension members:
- 1 From the Function Builder, click **Select Member**.
- 2 From **Dimension**, select a dimension.
- 3 Optional: To filter the list of members, select a member list from the Filter drop-down list.
- 4 Optional: For advanced filtering, highlight a member without selecting its check box, then select a member list.

This uses the highlighted member as the filter parameter. For example, if you highlight United States and change the filter list to Children, the system displays the children of United States.

- 5 Optional: To display dimension descriptions instead of labels, select Use Descriptions.
- 6 Optional: To display Dynamic Time Series members for a time dimension, select Dynamic Time Series Members.

Note:

The Dynamic Time Series option is available only with Essbase data sources.

7 Optional: To display active entities only, select Active Members.

Note:

Active Members is available only if the application has been set up for Organization by Period. Active Members is unavailable for Hyperion Enterprise data sources.

8 Select each member that you want to use.

Tip:

To deselect all selected members, click Select None.

- 9 Click to move the selected members to the **Selection** list. (to remove members from the Selection list, click or
- 10 Optional: To search for members in the selected dimension, click
- 11 When you finish selecting members, click **OK**.

HsGetValue

The HsGetValue function retrieves data from a data source for the dimension members that you specify. You can specify all dimension members. Any member not specified is considered a default POV member and uses the default in the POV Manager.

When users select Refresh or Refresh All, only HsGetValue is called. When users select Submit, HsSetValue is called first, HsGetValue is then called only if HsSetValue returns successfully.

The syntax is as follows:

```
HsGetValue ("Connection", "POV")
```

Here are two examples:

The function in the following example returns the value from the HFM01 application for the default Point of View.

Example 1:

```
HsGetValue("HFM01";"Scenario#Actual;Year#2004;Period#July;View#YTD;
Entity#UnitedStates.Connecticut;Value#USD;Account#Sales;ICP#[ICP
None];Custom1#GolfBalls;Custom2#Customer2;Custom3#[None];Custom4#
Increases")
```

The function in the following example returns the value from the HE application for the default Point of View.

Example 2:

```
HsGetValue("HE01";"Scenario#Actual;Period#July;View#YTD;
Entity#UnitedStates.Connecticut;Account#Sales")
```

HsSetValue

HsSetValue sends a data value from a worksheet to a data source for the dimension members that you specify. You can select any dimension member.

The syntax is as follows:

```
HsSetValue (dollar amount, "Connection", "POV")
```

Here are two examples:

The function in the following example sends the value from the H4 cell to the HFM01 application.

Example 1:

```
HsSetValue(H4, "HFM01", "Scenario#Actual; Year#2004; Period#"&B
$2&"; View#<Scenario View>; Entity#UnitedStates. Connecticut; Value#<Entity
Currency>; Account # "&$A4&"; ICP # [ICP
None];Custom1#GolfBalls;Custom2#Customer2;Custom3#[None];Custom4#
Increases")
```

The function in the following example sends the value from the H4 cell to the HE application.

Example 2:

```
HsSetValue(H4, "HE01", "Scenario #Actual; Period # "&B$2&"; View # < Scenario
View>;Entity#UnitedStates.Connecticut;Account#"&$A4")
```

Note:

To send data to a data source, you must have the appropriate load rule and write access for the data source.

HsCurrency

HsCurrency retrieves the currency value of the specified dimension member. Entity and Value are the only valid members for the HsCurrency function. Hyperion Enterprise does not support the Value dimension member.

The syntax for Financial Management is as follows:

```
HsCurrency ("Connection, Entity; Value")
```

Note:

The syntax for Hyperion Enterprise is the same as that of Financial Management with the exception of Value dimension member.

The following lines show an example of retrieving the entity currency where the currency for the East Sales entity is USD, and the currency for the UKSales entity is GBR.

```
HsCurrency("Comma", "Entity#EastRegion.EastSales; Value#<Entity Currency>.")
HsCurrency("Comma", "Entity#EastRegion.UKSales; Value#<Entity Currency>.")
```

In this example, the EastSales entity displays USD, and UKSales displays GBR.

HsDescription

HsDescription displays the description of the specified dimension member. You must specify the dimension members one at a time.

The syntax is as follows:

```
HsDescription ("Connection", "Dimension#Member")
```

For example, the following function displays the description for Custom 4.

HsDescription("HFM01","Custom4#Increases")

HsLabel

HsLabel displays the default member label for a specified dimension member.

The syntax is as follows:

```
HsLabel ("Connection, Dimension#")
```

For example, the following function retrieves the label for the Scenario dimension in the Comma application:

```
HsLabel ("Comma", "Scenario#")
```

HsGetText

HsGetText retrieves cell text from a data source. You can use all dimension members, or use cell references, the default Point of View, or a combination of all three.

The syntax is as follows:

```
HsGetText ("Connection", "POV")
```

For example, the following function returns the cell text from the HFM01 data source for the default Point of View.

```
HsGetText("HFM01", "Scenario#Actual;Year#2004;Period#"&B$2&";View#
<Scenario View>;Entity#UnitedStates.Connecticut;Value#<Entity
Currency>;Account#"&$A3&";ICP#[ICP
None];Custom1#GolfBalls;Custom2#Customer2;Custom3#[None];Custom4#
Increases")
```

HsSetText

HsSetText sends cell text to a data source. You can use all dimension members, or use cell references, the default Point of View, or a combination of all three.

The syntax is as follows:

```
HsSetText("Cell Text Comments", "Connection; POV")
```

For example, the following function sends the text from the H3 cell to the HFM01 application.

```
HsSetText("H3","HFM01;Scenario#Actual;Year#2004;Period#"&B
$2&";View#<Scenario View>;Entity#UnitedStates.Connecticut;Value#<Entity
Currency>;Account#"&$A3&";ICP#[ICP
None];Custom1#GolfBalls;Custom2#Customer2;Custom3#[None];Custom4#
Increases")
```

Accessing Functions Using Smart Tags

Smart tags is a Microsoft Office 2003 feature designed to save time by performing actions in Office that you would otherwise have other programs do. Smart tags recognizes specific keywords that invoke associated functionality. Smart View supports smart tags in Excel, Word, Power Point, and Outlook 2003. Using smart tags, you can type the "smartview" keyword in Excel, Word, Power Point, or Outlook to invoke a Smart View menu. Smart tags provide an alternative way of accessing functions for Financial Management, Hyperion Enterprise, and Essbase. You can also use smart tags to import BI+ content. See "Using Smart Tags to Import Reporting and Analysis Documents" on page 181. The "smartview" keyword is not casesensitive.

Note:

Smart tags are supported in Outlook 2003 only if the email editor is using Microsoft Word. Smart tags are not supported in Office versions before Microsoft Office 2003.

You can perform the following tasks in Smart View using smart tags:

- "Retrieving Values" on page 192
- "Retrieving Cell Text" on page 193
- "Retrieving Entity Currency" on page 193 •
- "Displaying the POV Description" on page 194
- "Removing Smart Tags for a Single Instance" on page 195
- "Stopping Hyperion Smart Tag Recognition" on page 195 •
- "Recognizing Smart View Smart Tags Again" on page 196

Note:

By default, smart tags are disabled. To enable smart tags, go to Tools, then AutoCorrect Options, then Smart Tags tab and select Label text with smart tags.

Retrieving Values

Using smart tags, you can retrieve a single data value using the HsGetValue function from Financial Management, Hyperion Enterprise, and Essbase.

- To retrieve a single value using smart tags:
- 1 Select **Hyperion, then Data Source Manager** to connect to a Hyperion Financial Management, Hyperion Enterprise, or Essbase data source. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.

See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.

2 Type smartview anywhere in the document, then move the mouse over the word.

The Smart Tags Action icon is displayed.

- 3 Click to display the **Smart View** menu.
- 4 Select Functions, then connection name, then HsGetValue.

The Member Selection dialog box is displayed.

Follow the directions for "Selecting Members for Functions" on page 188.

The value is displayed in your document.

Retrieving Cell Text

Using smart tags, you can retrieve cell text using the HsGetText function from Financial Management.

Note:

Essbase and Hyperion Enterprise does not support the HsGetText function.

- To retrieve cell text using smart tags:
- 1 Select Hyperion , then Data Source Manager to connect to a Hyperion Financial Management data source. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.
- 2 Type smartview anywhere in the document, then move the mouse over the word.
- 3 Click to display the Smart View menu.
- Select Functions, then connection name, then HsGetText.

The Member Selection dialog box is displayed.

Follow the directions for "Selecting Members for Functions" on page 188.

The cell text is displayed in your document.

Retrieving Entity Currency

Using smart tags, you can retrieve the entity currency for the selected members using the HsCurrency function from Financial Management and Hyperion Enterprise.

Note:

Essbase does not support the HsCurrency function.

- To retrieve the entity currency using smart tags:
- 1 Select Hyperion, then Data Source Manager to connect to a Hyperion Financial Management or Hyperion Enterprise data source. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.

See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.

- 2 Type smartview anywhere in the document, then move the mouse over the word.
- 3 Click (3) to display the Smart View menu.
- 4 Select Functions, then connection name, then HsCurrency.

The Member Selection dialog box is displayed.

5 Follow the directions for "Selecting Members for Functions" on page 188.

The entity currency is displayed in your document.

Displaying the POV Description

Using smart tags, you can display the description for the default Point of View member using the HsDescription function from Financial Management and Hyperion Enterprise.

Note:

Essbase does not support the HsDescription function.

- To display the POV description using smart tags:
- 1 Select Hyperion, then Data Source Manager to connect to a Hyperion Financial Management or Hyperion Enterprise data source. For Office 2007, click Connect in the Connections section of the Hyperion ribbon.

See "Adding a Data Source Through Direct Connection" on page 36 and "Adding a Data Source Through Shared Services" on page 38.

- 2 Type smartview anywhere in the document, then move the mouse over the word.
- 3 Click 1 to display the Smart View menu.
- Select Functions, then connection name, then HsDescription.

The Member Selection dialog box is displayed.

5 Follow the directions for "Selecting Members for Functions" on page 188.

The POV description is displayed in your document.

Removing Smart Tags for a Single Instance

If you want to use "smartview" as a word in your document, rather than as a smart tag, you can disable the smart tag icon or that instance. When you later type "smartview" in the document, the smart tag is still recognized.

- To remove smart tags:
- 1 Type smartview anywhere in the document, then move the mouse over the word.
- 2 Click 1 to display the Smart View menu.
- Select Remove this Smart Tag.

Stopping Hyperion Smart Tag Recognition

If you no longer want Office to recognize Hyperion smart tags, you can turn them off. You can turn off smart tags for a data type (for example, all Hyperion smart tags) or individual data items (specific keyword).

You can stop recognizing the Hyperion smart tag in two ways:

- Accessing the Smart View menu item
- Modifying the smart tag options
- To stop recognizing Hyperion smart tags directly through the Smart View menu:
- 1 Type smartview anywhere in the document, then move the mouse over the word.
- 2 Click to display the Smart View menu.
- Select Stop Recognizing "smartview", then select one of the following options:
 - as Hyperion® to prevent Word or Power Point from recognizing all Hyperion smart tags
 - as Smart Tag to stop recognizing the "smartview" keyword

Microsoft Office stops recognizing "smartview" as a smart tag. You must edit the XML exceptions file to add "smartview" back as a smart tag. See "Recognizing Smart View Smart Tags Again" on page 196.

- To stop recognizing Hyperion smart tags using smart tag options:
- 1 Type smartview anywhere in the document, then move the mouse over the word.
- 2 Click to display the **Smart View** menu.
- **Select Smart Tag Options.**

The AutoCorrect dialog box with the Smart Tags tab is displayed.

4 In Recognizers, clear Hyperion® (Hyperion® Smart View recognizer).

Note:

The AutoCorrect dialog box has other smart tag options. See the Microsoft Office documentation for more information.

5 Click OK.

Microsoft Office stops recognizing "smartview" as a smart tag. You must edit the XML exceptions file to add "smartview" back as a smart tag. See "Recognizing Smart View Smart Tags Again" on page 196.

Recognizing Smart View Smart Tags Again

When you stop recognizing a smart tag, the keyword is permanently deleted as a smart tag. To enable Microsoft Office to recognize a smart tag again, you must edit an XML file that contains all the exceptions. Usually, this XML file is located at a user's directory:

<Drive letter>:\Documents and Settings\<username>\Application Data \Microsoft\Smart Tags\Exceptions

You can find more information about editing the XML exceptions file at:

http://msdn.microsoft.com/library/en-us/stagsdk/html/stconExceptionList.asp

Creating Functions Manually

When you create functions in worksheets manually, you must precede each function with an equal sign (=). In each function, you can specify a connection and a full or partial Point of View. If you do not specify a connection, the system uses the default connection. Some functions also require a value. For example, the HsSetValue function requires that you set a value in addition to the connection and Point of View.

The connection, if specified, must precede the Point of View. The Point of View is made up of dimension#member pairs, where the pound symbol (#) is used as a separator between the dimension and member, for example, Entity#Connecticut. Parent-child relationships can be represented with a period, for example, Entity#UnitedStates.Maine.

The connection and Point of View can be grouped as one parameter, for example "My_connection;Entity#UnitedStates". Alternatively, they can be split up into several function parameters, for example, "My_connection", "Entity#UnitedStates", "Account#Sales". If the connection and Point of View are in the same function parameter, the connection and each dimension#member pair are separated by a semi-colon (;) character, for example, "My_connection;Entity#UnitedStates;Account#Sales".

Note:

Functions can contain a maximum of 256 characters.

Example:

=HsGetValue("HFM01","Scenario#Actual;Year#2004;Period#July;View#YTD; Entity#UnitedStates.Connecticut;Value#USD;Account#Sales;ICP#[ICP None];Custom1#GolfBalls;Custom2#Customer2;Custom3#[None];Custom4# Increases")

- To create functions manually:
- 1 Select the cell in which you want to create the function.
- 2 Enter the function, then select the parameters for the function.
- 3 To refresh the worksheet, select Hyperion, then Refresh

Editing Functions

You can edit a previously-defined function to change the dimension members.

To change the default Point of View, you use the POV Manager. See "Using the POV Manager" on page 221.

- To edit a function:
- 1 Open a worksheet and select the cell for which you want to edit the function.
- 2 Select Hyperion, then Functions, then Function Builder. For Office 2007, click Function Builder in the Task section of the Hyperion ribbon.

The function is displayed in the Function dialog box.

- 3 Click Select Member, then from the list of dimension members, select the members that you want to use.
- 4 To add another function, click **Add to Function**.

The function is added after the currently displayed function. To replace the current function or clear all functions, click Clear.

- 5 To validate the function syntax before pasting it to the worksheet, click **Validate**.
- 6 Click **OK** to paste the function to the worksheet.
- 7 To refresh the worksheet, select **Hyperion**, then **Refresh**. For Office 2007, click **Refresh** in the **Review** section of the **Hyperion** ribbon.

Running Functions

After you have created functions, you can run the functions to retrieve the current values automatically and update the worksheet.

- To run functions:
- 1 Open the worksheets for which you want to run functions.
- 2 Select one of the following:
 - To run functions and update all worksheets, select Hyperion, then Refresh All

To run functions and update only the active worksheet, select Hyperion, then Refresh.

Common Function Error Codes

Some common error codes displayed in functions:

#NO CONNECTION - You are not connected or logged on to a data source.

#INVALID - Invalid metadata. These cells are displayed in red in data grids. Invalid cells that contain a value display the value as zero.

#LOCKED - The cell is locked.

#NO ACCESS - You do not have access to this cell.

#NO DATA - The cell contains NoData. You can select to display zeros instead of NoData. Cells use the Replacement text that you specify in the Options dialog box.

#INVALID INPUT - The HsSetValue data value is not valid, for example, a text string.

#READ ONLY - This is for the HsSetValue function only when the cell is Read-only.

#NO ROLE ACCESS - You do not have the Financial Management LoadExcelData security role.

#NEEDS REFRESH - Data needs to be refreshed.

#INVALID DIMENSION - An invalid dimension is specified in the function.

#INVALID MEMBER - An invalid dimension member name is specified in the function.

#NAME - Microsoft Excel doesn't recognize text in a formula. When you forward a spreadsheet that contains functions to a user who does not have Smart View, users can view the same data as the functions remain displayed on the spreadsheet. When the user edits the function or selects Refresh, the function changes to #Name.

Working with Offline Planning

In This Chapter

| About Working Offline | 199 |
|--|-----|
| Taking Data Forms Offline | |
| Refreshing the Offline Data Form Definition and Data | 203 |
| Entering Data with Menus. | 205 |

About Working Offline

Working disconnected from the Planning server, planners can work much as they do online. After taking data forms offline, planners can adjust data, view instructions, add supporting detail, and create data manipulation formulas, then save the changes to the Planning server. Thus, planners can work with plans and forecasts in any location, such as on planes, hotels, or home, with or without an Internet connection.

Typical offline working cycle:

- Create an offline connection (see Chapter 4, "Connecting to Data Sources").
- Select and download the data forms, dimensions, and members "Taking Data Forms Offline" on page 200.
- Close Excel.
- At your destination, start Excel, open the offline connection you created, and work with the data forms, dimensions, and members that you took offline.
- When you are done, save the changed data to the Planning server (see "Synchronizing Data to the Planning Server" on page 202).

Follow these guidelines when using data forms offline:

- Synchronize data with the server only once during an offline session to save processing time. Each time you select Sync Back To Server, all data changed within the data forms taken offline since the beginning of the session is saved to the server.
- Currency conversion is not supported offline.
- If you lose a row or column of data when you refresh a data form, contact the administrator.
- After you reconnect to the server, check that the work you completed offline is correct in the database.

Taking Data Forms Offline

Note:

You can include both online and offline data forms in the same Excel workbook.

To take data forms offline:

- In Excel, establish a connection to the online Planning data source that contains the preferred data form or forms (see "Adding a Data Source Through Direct Connection" on page 36 for instructions).
- Select Hyperion, then Forms, and then Take Offline. For Office 2007, click Take Offline in the Forms section of the Hyperion ribbon.

The Take Offline Wizard is displayed; all data forms that you can take offline are listed.

Note:

If Take Offline is not available, see the Planning administrator.

- From the Available Forms/Folders list, select or browse for data forms.
 - Click the expand (+) and collapse (-) signs next to the folder and data form names to expand or collapse the list.
 - To select a folder or data form, select the check box next to its name.
 - If you clear the check box next to a folder name, no data forms in that folder are selected.

Click Next.

Double-click a dimension.

If you selected multiple data forms, the dimensions displayed are merged from the dimensions available for the selected data forms. Note that you can select only one dimension.

Select members and system variables from the Member Selection page.

The list contains members and system variables of the selected dimension.

- To select all members, select the check box next to the dimension name.
- Click the plus or minus sign to expand or collapse the list.
- To search for a member or system variable in the dimension hierarchy, click. For Search, type part or all the member or system variable name and click or 🙀 or

To Move members or system variables to or from the Selected Members list:

- Click to add selected members or system variables.
- Click to remove selected members or system variables.
- Click to remove all members or system variables.
- Click the arrow to the right of to list the available member relationships.

About member relationships:

Table 12 Member Relationships

| Relationship | Members Included on the Data Form |
|---------------------|--|
| Member | The selected member |
| Descendants | All members below the selected member |
| Descendants (inc) | The selected member and all its descendants |
| Ancestors | All members above the selected member |
| Ancestors (inc) | The selected member and all its ancestors |
| Siblings | All members from the same level in the hierarchy as the selected member, excluding the selected member |
| Siblings (inc) | The selected member and all its siblings |
| Parents | The member in the level above the selected member |
| Parents (inc) | The selected member and its parent |
| Children | All members in the level immediately below the selected member |
| Children (inc) | The selected member and all its children |
| Level 0 Descendants | All descendants of the selected member that have no children |

Note:

Different data forms may have children and page-member selections. The Page drop-down list should contain at least one member for each data form from each dimension.

- 7 Click OK.
- Repeat steps 5-7 to select members or system variables for each dimension in the list.
- Click Next.
- 10 Supply a unique name and a description for the offline connection (for example, you may want to describe the types of data forms included).
- 11 Click Finish to download the selected data forms and members.
- 12 Click **OK**, then click **Done**.

Working with Data Forms Offline

For an overview of working offline, see "About Working Offline" on page 199.

- To work with data forms offline:
- 1 Launch Excel and select Hyperion, then Data Source Manager. For Office 2007, click Connect in the **Connections** section of the **Hyperion** ribbon.
- 2 Select the offline connection.

Online connections specify *Planning* in the Provider column; offline connections specify *Offline* Planning.

- 3 Click Connect.
- 4 Select Hyperion, then Forms, then Select Form to select a data form. For Office 2007, click Select Form in the Forms section of the Hyperion ribbon.

Note:

If you have a data form open while you are directly connected to the Planningserver, then take the data form offline in the same session, you must reopen the data form from the offline connection to work with it offline.

- 5 In the offline data form add or change data.
- 6 Select Hyperion, then Submit Data. For Office 2007, click Submit in the Review section of the Hyperion ribbon.

The changed data is saved locally. You can exit Excel without losing the changed data.

Synchronizing Data to the Planning Server

- To save changed data to the Planning server for all data forms and members taken offline:
- 1 Select Hyperion, then Forms, and then Sync Back To Server. For Office 2007, click Sync Back to Server in the Forms section of the Hyperion ribbon.

The Sync Back To Server option is available only when the current connection is an offline connection.

- 2 Logon to the Planning server.
- 3 Click Sync Back All.
- 4 Click OK.
- To save changed data to the Planning server for selected data forms and members taken offline:
- 1 Select Hyperion, then Forms, and then Sync Back To Server. For Office 2007, click Sync Back to Server in the Forms section of the Hyperion ribbon.

The Sync Back To Server option is available only when the current connection is an offline connection.

- 2 Logon to the Planning server.
- 3 Click Next.

- Double-click a dimension.
- Select members and system variables from the Member Selection page.

The list contains members and system variables of the selected dimension.

- To select all members and system variables, select the check box next to the dimension name.
- Click the plus or minus sign to expand or collapse the list.
- To search for a member or system variable in the dimension hierarchy, click. For Search, type part or all the member or system variable name and click or

To Move members and system variables to or from the Selected Members list:

- Click to add selected members or system variables.
- Click to remove selected members or system variables.
- Click to remove all members or system variables.
- 6 Click OK.
- Repeat steps 4-6 to select members or system variables for each dimension in the list.
- Select Finish to save data.
- Click OK and Done.

Refreshing the Offline Data Form Definition and Data

When you refresh the offline data form definition, you:

- Update data on the offline data forms with current values from the online data forms.
- Add or delete members or data forms from the ones available during an offline session.
- To update offline data and the offline data form definition:
- 1 If necessary, launch Excel and select Hyperion, then Data Source Manager. For Office 2007, click **Connect** in the **Connections** section of the **Hyperion** ribbon.
- 2 Select the connection associated with the current offline session.

For information, see "About Independent Provider Connections" on page 35.

3 Select Hyperion, then Forms, and then Refresh Offline Definition. For Office 2007, click Refresh Offline in the Forms section of the Hyperion ribbon.

Note:

If you are using an offline connection and the Refresh Offline Definition option is not available, contact the Planning administrator. This option is not available if you are using an online connection.

Enter the user name and password for the online data source.

Because you want to refresh the offline data from the Planning server, you must log on to the server.

Take an action:

Click Refresh All to update all members and data forms taken offline with current online values and definitions. Skip to Step 10.

Note:

Refresh All maintains the current offline data form definition. You change the offline data form definition if you do not select Refresh All. The members and data forms that you refresh remain part of the offline data form definition; the members and data forms that you do not select are no longer part of the definition. For example, if you take five data forms offline, then select only two of them when you refresh the offline data form definition, the three data forms you do not select are no longer available offline.

- Click Next to select which data forms, members, and system variables to update.
- Double-click a dimension.
- Select members and system variables from the Member Selection page.

The list contains members and system variables of the selected dimension.

- To select all members, select the check box next to the dimension name.
- Click the plus or minus sign to expand or collapse the list.
- To search for a member or system variable in the dimension hierarchy, click. For Search, type part or all the member or system variable name and click or

To Move members and system variables to or from the Selected Members list:

- Click to add selected members or system variables.
- Click to remove selected members or system variables.
- Click to remove all members or system variables.
- Click OK.
- Repeat steps 6-8 to select members or system variables for each dimension in the list.
- 10 Click Finish to start the refresh.
- 11 Click OK, then click Done when the refresh is complete.

When you connect to the Planning server to synchronize offline data to it, your access privileges are checked to ensure that your read-write rights have not changed since you went offline. This check largely involves verifying your access. It also tracks changes to metadata, data form definition, security, and Planning Unit ownership.

Entering Data with Menus

The administrator can set up data forms that include menus. Using menus, you right-click a member and select a menu item to open a URL, data form, workflow, or business rule. For example, a menu item can open another data form to let you drill down into the data, go to another scenario and version in the planning unit, or launch a calculation.

- To enter data with menus:
- 1 Open a data form containing a menu.
- 2 Click a row member, and select a menu option from the list.

The values in the list depend on how your administrator sets up this feature. If the menu includes submenus, you can select a value from the submenu. The menu can also be set up on a column member.

Depending on the action that was performed by the menu item, you can continue your work on the Web page, data form, or workflow.

If a business rule was launched that includes a runtime prompt, enter the required information. See "Entering Runtime Prompts for Business Rules" on page 70.

Your administrator can customize the runtime prompt that displays for menus. For example, it can be displayed with a Classic or Streamlined interface, and the Launch button can use another name, such as OK.

16

User Preferences

In This Chapter

| Ad Hoc Options | 207 |
|---------------------|-----|
| Display Options | 212 |
| Setting Cell Styles | 218 |

This section explains user preference settings in the Options dialog box. Depending on how your Administrator has set up Smart View, some options may not be available.

Ad Hoc Options

The ad hoc options control how data is retrieved into the spreadsheet and which data is retrieved in ad hoc queries.

The following topics describe the tasks you can perform on the Ad Hoc tab of the Options dialog box:

- "Suppressing Types of Data" on page 207
- "Indenting Member Names" on page 208
- "Navigating Without Data" on page 208
- "Double-clicking in Ad Hoc Operations" on page 209
- "Undo and Redo" on page 209
- "Zoom Options in Connected Sheets" on page 210
- "Member Retention Options" on page 211
- "Ancestor Positions in Hierarchies" on page 211

Note:

Options dialog box settings are applied globally to all sheets in the workbook.

Suppressing Types of Data

You may want to prevent Smart View from displaying rows in Ad Hoc grids that contain certain types of data.

- To suppress rows:
- 1 Click Options or select Hyperion then Options, then Ad Hoc.
- 2 Under Suppress Rows, select any of the following options:
 - No Data/#Missing to suppress rows containing cells for which no data exists in the database (no data is not the same as zero)
 - If you later deselect No Data/#Missing, suppressed values are returned only from that point on. You must drill up then drill down on a member to retrieve values that were suppressed while this option was selected.
 - Zero to suppress rows that contain only zeroes
 - No Access to suppress rows that contain data that you do not have the security access to view
 - Invalid to suppress rows that contain invalid values
 - Underscore Characters to suppress underscore characters in member names (Not available in Essbaseor Smart Slice operations)
 - Repeated Members to suppress repeated member names (This is the only suppress option available in Oracle BI EE.)
- Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Indenting Member Names

(Not available for Oracle BI EE) Indenting member names makes it easier to view relationships between members in the spreadsheet.

- To indent member names in the spreadsheet:
- 1 Select Hyperion then Options, then Ad Hoc. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Under **Indentation**, select one of the following:
 - None
 - **Subitems** to indent only descendants. Ancestors are left-justified in the column
 - Totals to indent only ancestors. Descendants are left-justified in the column
- Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Navigating Without Data

(Not available for Oracle BI EE) You can speed up navigation operations such as Pivot, Zoom, Keep Only and Remove Only by preventing the calculation of source data while you are navigating.

- To navigate through the sheet without retrieving data:
- Select Hyperion then Options, then Ad Hoc. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Under Mode, select Navigate without Data.
- 3 Click OK. The setting takes effect after you refresh or perform a drill operation.
- 4 When you are ready to retrieve data, clear Navigate without Data.

Double-clicking in Ad Hoc Operations

(Not available for Oracle BI EE) Normally, double-clicking in an Excel cell puts its contents in edit mode, but you can change it to Smart View Ad Hoc functionality for Essbase, Financial Management, or Hyperion Enterprise data sources (double-clicking in Planning retains standard Planning functionality).

Enabled for Ad Hoc operations, double-clicking in a blank spreadsheet connected to an Essbase, Financial Management, or Hyperion Enterprise data source retrieves the default grid from the server. Once the grid is loaded, double-clicking on members drills down or up to more or less detail.

Note:

If Essbase Spreadsheet Add-in and Smart View are installed on the same computer and you have not completed the steps in "Smart View and Essbase Spreadsheet Add-in" on page 28, double-clicking invokes the login dialog box for Essbase Spreadsheet Add-in. This behavior is applicable only for Spreadsheet Add-in in Essbase Release 7.1.2.

- To enable double-clicking for ad hoc operations:
- Select Hyperion then Options, then Ad Hoc. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Under Mouse Operation, select Use Double-click for Ad Hoc Operations.
- 3 Click OK. The setting takes effect after you refresh or perform a drill operation.

Undo and Redo

You must enable Undo and Redo functions before they are available from the Hyperion menu. Smart View Undo and Redo behave differently depending on the data source to which you are connected.

Using Ad Hoc analysis with Essbase, Financial Management, Oracle BI EE, or Hyperion
Enterprise data sources: Undo undoes Zoom In, Zoom Out, Keep Only, Remove Only, or
Refresh command and restores the previous database view to the worksheet. Performing an
Undo after modifying member data returns the sheet to its state before the last refresh, not
to its state before the data modification.

Using data forms with Financial Management, Hyperion Enterprise, or Planning: Undo undoes the last user action in a cell. Using Undo and Redo you can go backward and forward through actions that were performed in the data form until you submit data to the server.

Note:

You cannot undo operations that are performed on the server rather than in Smart View, such as calculation status.

Enabling Undo and Redo

- When you enable Undo, you are also enabling Redo. To enable Undo:
- 1 Select Hyperion then Options, then Ad Hoc. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Under Undo, select Enable.
- 3 In Number of Undo Actions, specify the number of permissible Undo operations 2 through 100. This is also the number of **Redo** operations allowed.
- Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Using Undo and Redo

Select Hyperion, then Undo or Redo to undo or redo operations. You can undo and redo as many times as specified on the Options dialog box Ad Hoc tab.

Zoom Options in Connected Sheets

(Not available for Oracle BI EE) The Zoom group controls the behavior of all Zoom In operations except within the selected group.

- To specify zoom options in the connected sheets:
- 1 Select Hyperion then Options, then Ad Hoc. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Under Zoom-In, select one of the following:
 - Next Level to retrieve data for the children of the selected members (default)
 - All Levels to retrieve data for all descendants of the selected members
 - **Bottom Level** to retrieve data for the lowest level of members in a dimension
 - Sibling Level to retrieve data for the siblings of the selected members (unavailable for Financial Management data sources)
 - Same Level to retrieve data for all members at the same level as the selected member or members
 - Same Generation to retrieve data for all members of the same generation as the selected member or members

• Formulas to retrieve data for all members that are defined in the formula of the selected member or members (can be a member equation or a consolidation to the parent)

Note:

Zoom-In options for Sibling Level, Same Level, Same Generation, and Formulas are not available if you are connected to a Hyperion Enterprise data source.

3 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Member Retention Options

(Not available for Oracle BI EE).

- To specify member retention in the connected sheets:
- 1 Select **Hyperion** then **Options**, then **Ad Hoc**. For Office 2007, click **Options** in the **Options** section of the **Hyperion** ribbon.
- 2 Select one or more of the following:
 - Include Selection to retain the selected member along with the other members retrieved as a result of the zoom
 - Include Selection is enabled by default.
 - Within Selected Group to apply only to the group of members in which the selection is made

 This setting is meaningful only when the sheet contains two or more dimensions of data
 down a sheet as rows or across a sheet as columns. (This setting also affects the behavior of
 Keep Only and Remove Only operations.)
 - Remove Unselected Groups to remove all dimension groups that are not in the selected group
- 3 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Ancestor Positions in Hierarchies

Ancestor position determines the order in which hierarchies display in the spreadsheet.

Note:

Ancestor Position options are not available for Essbase or Oracle BI EE data sources.

- To set the ancestor position for hierarchies:
- 1 Select **Hyperion** then **Options**, then **Ad Hoc**. For Office 2007, click **Options** in the **Options** section of the **Hyperion** ribbon.
- 2 Under Ancestor Position, select one of the following:
 - Top to display hierarchies in order from the highest level of the hierarchy to the lowest

For example, when you expand Sales, you see Gross Margin.

- Bottom to display hierarchies in order from the lowest level of the hierarchy to the highest For example, when you expand Gross Margin, you see Sales.
- 3 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Setting the Provider Services Connection URL

Data source types: Essbase, Planning, Oracle BI EE

For data sources that connect to Smart View through Provider Services, you can manage the connection URL.

To change the default URL or use a URL different from the default, select Override default and modify the URL shown in the Default URL text box (see Chapter 4, "Connecting to Data Sources").

Display Options

The options on the Display tab control how data is displayed in the spreadsheet.

The following topics describe the tasks that you can perform on the Display tab of the Options dialog box:

- "Replacement Labels" on page 212
- "Cell Status" on page 213
- "Member Name Display Options" on page 214
- "UI Colors" on page 214
- "Format Strings" on page 214
- "Scale and Decimal Display Options" on page 215 •
- "Using Excel Formatting" on page 215
- "Formula Preservation After POV Changes" on page 216
- "Displaying and Logging Messages" on page 217

Replacement Labels

For Excel data cells whose data is missing, invalid, or inaccessible (because you do not have permission to view the data), you can specify either text or numeric zero (0) replacement labels. Text labels, #INVALID for example, are descriptive but they cause Excel functions to fail.

Numeric Zeroes

Numeric zeroes permit Excel functions; however, when you submit data, Smart View removes all zeroes from submission even if they are actual zeroes and not replacement labels. As a result, you cannot submit the actual value 0 with this option. The numeric zero option is available only in ad hoc, forms, and function-based grids.

When you specify numeric zero replacement labels:

- Excel formatting for the cell is retained.
- All calculations that are dependent on the cell compute correctly and take the value of the cell as zero.
- The numeric zero is for display only. When you submit, the zero value is NOT submitted back to the data source.
- To specify labels for missing or invalid data in Excel cells:
- 1 Select **Hyperion** then **Options**, then **Display**. For Office 2007, click **Options** in the **Options** section of the **Hyperion** ribbon.
- 2 Under Replacement, enter either #NumericZero to specify numeric zero replacement, or text of your choice in the fields:
 - #NoData/Missing Label: for missing data.
 - **#No Access Label:** for data that you do not have permission to view. (Not available for Oracle BI EE)
 - **#Invalid/Meaningless:** where the data in the intersection is invalid for valid metadata combinations.
- 3 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Note:

Replacement labels apply only to data cells, not to cells that contain dimensions or members.

Cell Status

(Not available for Oracle BI EE) In data grids, in addition to displaying actual data, you can choose to display the calculation or process status of the cells in the grid.

- ➤ To specify cell status:
- 1 Select **Hyperion** then **Options**, then **Display**. For Office 2007, click **Options** in the **Options** section of the **Hyperion** ribbon.
- 2 In the **Cell Status** group, select one of the following options:
 - Data to show actual data.
 - Calculation Status to show whether data needs to be calculated, translated, or consolidated (Hyperion Enterprise and Financial Management data sources only)
 - **Process Management** to show the process management level for combinations of data called process units (Financial Management data sources only)

Member Name Display Options

If your Essbase system permits the use of duplicate member names, you can choose whether to display member names with or without their descriptions or aliases.

- To select how member names are displayed:
- 1 Select Hyperion then Options, then Display. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Under **Member Name Display Options**, select an options:
 - Member Name Only to display fully-qualified names
 - Member Name and Description to display fully-qualified names and descriptions (aliases) in the same cell. (Hyperion Enterprise data source only)
 - Description Only to display aliases only. (Not available if connected to Hyperion Enterprise data source)

Note:

If you select Description Only in free-form mode, fully-qualified names are displayed initially. After you manually add, remove, or edit any comments and refresh, aliases are displayed.

3 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

UI Colors

(Not available for Oracle BI EE) You can specify one or two background colors for Smart View dialog boxes that contain rows of information.

- To select dialog box colors:
- 1 Select Hyperion then Options, then Display. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Select **UI Colors**. Two color buttons are displayed.
- 3 One at a time, click a color button and select a color.
- 4 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Format Strings

Data Sources: Essbase

Numerical data is sometimes more informative if it contains formatting or is presented as text. For example, the number -12000 would be better presented in a financial statement as (12, 000). In Smart View, such presentations of data are called "format strings" and are defined by the Administrator in Oracle Essbase Administration Services.

- To view format strings rather than numerical data (if they have been defined for you):
- 1 Select **Hyperion**, then **Options**.
- 2 On the Display tab under Format, select Format String
- 3 Click OK.

Scale and Decimal Display Options

(Not available for Oracle BI EE).

- To set scale and decimal options:
- Select Hyperion then Options, then Display. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 From Scale, select a scale value for data in the grid. Your selection results are displayed in the Value 1 will be text box.
- 3 From Decimal Places, select the number of decimal places to be displayed in the spreadsheet. Your selection results are displayed in the Value 1 will be text box. (Hyperion Enterprise, Planning, and Financial Management data sources only).
- 4 To use a thousands separator, select the **Use Thousands Separator** check box (Hyperion Enterprise, Planning, and Financial Management data sources only).

If you select this option, do not use # or \$ as the thousands separator in Excel's International Options.

Note:

The Use Excel Formatting option (described in "Using Excel Formatting" on page 215) overrides the Use Thousands Separator option when both are selected.

5 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Using Excel Formatting

(Not available for Oracle BI EE) You can let Smart View control cell formatting or you can control formatting by using Excel formatting functionality. Using Excel formatting is generally preferably for highly-formatted reports, and you *must* use Excel formatting for data sources whose application-specific colors are not supported by Excel's color palate.

When you use Excel formatting, Smart View does not reformat cells based on the grid operations you perform, and it does not mark cells as dirty when you change data values. Smart View does preserve the formatting on the worksheet between operations.

Use Excel Formatting is supported for refresh and submit actions but not for zoom or pivot operations.

The Use Excel Formatting option overrides the following:

- The Use Thousands Separator option on the Display tab of the Options dialog box
- Styles selected in the Cell Styles tab of the Options dialog box
- Styles set using Capture Formatting

Note:

Member and numeric formats may unexpectedly change after pivot operations. For example, member names may be centered and numeric values may be left justified. You can reset the grid to the proper format using the formatting options of Excel.

- To use Excel formatting:
- 1 Select Hyperion then Options, then Display. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Select Use Excel Formatting.
- 3 Click **OK**. The setting takes effect after you refresh the worksheet.

Formula Preservation After POV Changes

Select Preserve Formula on POV Change to preserve formulas after you change the POV. Upon refresh, formulas that have been entered into data forms are maintained; only the point of view of the formula changes.

Note:

The Preserve Formula on POV Change option applies only to data forms in Hyperion Enterprise, Planning, and Financial Management.

For example, suppose you inserted a formula into a data form grid, with a reference to cell B4 on Sheet 2 of the workbook. The entity dimension is currently set to "Fairfield" within the page POV:

```
=Sheet2!B4
```

Now suppose you want to apply the same formulas to the "Hartford" entity. Change the entity dimension within the page POV from Fairfield to Hartford and refresh the data form. This preserves the existing formula:

```
=Sheet2!B4
```

The data for Hartford is retrieved onto the data form.

Using the "Preserve Formula on POV Change" option has no impact on formulas in the data form; it simply retains the existing formulas within the data grid.

However, formulas will be lost under these circumstances:

If they reside on intersections that are invalidated due to suppression options set at the data source level.

If a cell changes from input level to Read Only, Invalid, and so on.
 For example, a valid cell reference made on one data form may reference a Read Only cell on another data form.

Note that when a formula is lost, it does not reappear on the data form. You must reenter it.

- To enable formula preservation after changes have been made to the POV:
- 1 Select **Hyperion** then **Options**, then **Display**. For Office 2007, click **Options** in the **Options** section of the **Hyperion** ribbon.
- 2 Select Preserve Formula on POV Change.
- 3 Click OK. The setting takes effect after you refresh or perform a drill operation.

Reducing Excel File Size

(Not available for Oracle BI EE) You can compress the metadata maintained in Excel files containing Smart View workbooks by selecting **Reduce Excel File Size** on the Display tab of the Options dialog box.

Displaying and Logging Messages

(Not available for Oracle BI EE) All error, warning and informational messages from the connected data source are displayed when they occur, but you can choose which of these message levels to record in a log file. The actual messages displayed depend on the provider (Hyperion Enterprise, Planning, Financial Management, Essbase, etc.) to which you are connected.

- To display and log messages:
- Select Hyperion then Options, then Display. For Office 2007, click Options in the Options section of the Hyperion ribbon.

Under Log Messages, select one of the following:

- Information to display all messages, including warnings and errors recommended to diagnose problems. Adversely impacts performance.
- Warnings to display warning and error level messages. Adversely impacts performance.
- Errors to display error messages only recommended for general use; has minimal impact on performance.
- None to suppress all messages.
- 2 If you want to keep a record of messages in a log file, select Route Messages to Log File.

Note:

Click the browse button to select a new location for the log file. Documents and Settings \"user name"\Local Settings\TEMP\SmartViewLogs.log is the default location.

- 3 Optional: To clear the log file each time Smart View is launched, select Clear Log File on Next Launch.
- 4 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Setting Cell Styles

Smart View data source providers each have a set of default cell styles to identify cell types. For example, Planning designates dirty cells by a yellow background and read-only cells by a gray background. You can modify default styles in Smart View change cell styles by selecting different fonts, background colors, or border colors.

Note:

For Oracle BI EE, you can set cell styles only for Members, Parent and Child.

- To set cell styles for member and data cells:
- 1 Select Hyperion then Options, then Display. For Office 2007, click Options in the Options section of the Hyperion ribbon.
- 2 Expand the list and select a member or data cell.
- 3 Click Properties and select Font, Background, or Border to modify the cell style.
- 4 Click **OK**. The setting takes effect after you refresh or perform a drill operation.
- 5 Optional: To revert cell styles to the default styles of the connected Smart View provider, click Default Styles.

Note:

Do not use Smart View worksheets and worksheets created by means other than Smart View in the same workbook; this causes colors in the non-Smart View sheets to change into unintended colors when you refresh.

Note:

The Use Excel Formatting option (described in "Using Excel Formatting" on page 215) overrides all style options set in the Cell Styles tab of the Options dialog box.

Cell Style Precedence

Cells can be associated with more than one style; you can specify the order in which styles are applied in such cells.

Note:

To see a child member style, turn off the parent member style. To see a shared member style, turn off both parent and child member styles.

- To change the order of precedence for member and data cell styles:
- 1 Select **Hyperion** then **Options**, then **Display**. For Office 2007, click **Options** in the **Options** section of the **Hyperion** ribbon.
- Select the member or data cell and then click Move Up or Move Down to give higher or lower precedence, respectively, to the style for that cell type.
- 3 Click **OK**. The setting takes effect after you refresh or perform a drill operation.

Note:

To revert cell precedence of cell styles to their defaults, click Default Styles.

Using the POV Manager

In This Chapter

| Editing the Point of View | 221 |
|---|-----|
| Selecting Dimension Members for the POV | 222 |
| Copying and Pasting a POV | 223 |
| Deleting a POV | 224 |
| Printing POV Members in Header and Footer | 224 |

When you select a Point of View (POV), the system retrieves and calculates values for the Point of View dimension settings from the data source unless you specify different settings. You can use the POV Manager to edit a POV, save it to a workbook, copy and paste a saved Point of View, or delete a Point of View from a workbook.

Note:

Changing POV using POV Manager isn't supported for Oracle BI EE.

Editing the Point of View

Connections have a default POV. You can select dimension members for the default connection POV. When you select the connection in the POV Manager, the details of the connection and the dimension selections or defaults are displayed.

- To edit the Point of View:
- 1 Click POV Manager or select Hyperion, then POV Manager.
- Expand the POVs list.
- 3 From the Active POV list, select the active connection for which you are changing the Point of View.
- Click the Member Selector button and select the dimension members that you want to use for the Point of View.

For more information, see "Selecting Dimension Members for the POV" on page 222.

The worksheet is refreshed with the Point of View changes.

- Click Close.
- 6 To refresh the worksheet, click **Refresh** or select **Hyperion**, then **Refresh**.

Selecting Dimension Members for the POV

In the POV Manager, you can select dimension members to use as a default POV for an Ad Hoc grid, or for the background POV for dimensions when using functions. From the POV Manager, you can select only one member per dimension.

Note:

Any changes to the background POV are not reflected in the cell until a refresh is completed.

To change the starting POV for an Ad Hoc grid, you must first open the POV Manager, select members for the POV, and then start the Ad Hoc grid. For example, in a Financial Management data source, the top member, which is None, is used by default. You might want to change Account to start with Gross Margin, and change Entity to start with North America.

Note:

After the Ad Hoc grid has been started, you can no longer make changes through the POV Manager, but instead make changes using the member selector in the Ad Hoc grid.

You use the Member Selection dialog box to select members. You can filter the list of members and you can view labels or descriptions for dimension members. For example, when you select to view descriptions, the P Series member is displayed as "P Series - Phones and PDAs".

You can also search for dimension members. For more information, see step 10 on page 223.

- To select dimension members:
- 1 From the POV Manager, click **Member Selector**.
- 2 From the **Dimension** drop-down list, select a dimension.
- 3 Optional: To filter the list of members, select a member list from the Filter drop-down list. See "Filtering Members" on page 98.
- 4 Optional: For advanced filtering, highlight a member without selecting its check box, then select a member

This uses the highlighted member as the filter parameter. For example, if you highlight United States and change the filter list to Children, the system displays the children of United States.

- 5 Optional: To display dimension descriptions instead of labels, select Use Descriptions.
- 6 Optional: To display Dynamic Time Series members for a time dimension, select Dynamic Time Series Members.

Note:

The Dynamic Time Series option is available only when working with Essbase data sources.

7 Optional: To display active entities only, select Active Members.

Note:

The Active Members option is available only if the application has been set up for Organization by Period. For information on Org by Period, see the Oracle Hyperion Financial Management, Fusion Edition Administrator's Guide.

8 Select the check box next to each member that you want to use.

Tip:

To deselect members, click Select None.

- Click Add to move the selected members to the Selection list.
- 10 Optional: To search for members in the selected dimension, use the Search button.
- 11 When you finish selecting members, click **OK**.

Copying and Pasting a POV

You can copy a POV that has been saved in a workbook and paste it to another workbook. For example, suppose another user has created a worksheet with functions and saved the connection POV to the workbook. They can send you the workbook and to ensure that you're looking at the same data, you can use the saved Point of View.

Saved POVs are referenced with the data source name of the computer on which they were saved. When you use a saved POV, you must have the same data source as the one for the saved POV.

For example, suppose someone saves a POV to a workbook using the Comma data source and referencing the Comma application, then sends it to you with instructions to use the saved POV. You must have a data source named Comma that references the Comma application/cube at the same URL and server to use the saved POV. If the data sources do not match, or if one references a different application or uses a different Web or application server, you cannot paste the POV into the active worksheet.

When using the POV manager to select POV values for ad hoc analysis, you must copy the POV defined in the POV Manager to a worksheet that is not associated with a connection, otherwise the POV has no effect. When you copy the POV to an unconnected sheet and then refresh, the POV values are used.

- To copy and paste a Point of View:
- 1 Click POV Manager or select Hyperion, then POV Manager.
- 2 From the **Active** folder, select the Active application connection.
- 3 Select the members for the POV and save the workbook.
- 4 Click Copy.
- Expand the Saved folder to select the workbook and worksheet into which you want to paste the POV, and click Paste.

Tip:

You can also drag and drop a POV to copy and paste it. For example, from the Saved POV list, you can select a POV from a workbook, and then drag it to the Active POV list. Or, conversely, you can select a POV from the Active POV list, and then drag it to the Saved POV list.

- 6 Save the workbook.
- 7 Optional: To copy a saved POV from another workbook, open the workbook and select the saved POV.

The Saved folder displays your source workbook and the new workbook.

- 8 Copy and paste the source POV to the target worksheet in the **Saved** folder.
- 9 To close the POV Manager, click Close.
- 10 To refresh the worksheet, click Refresh or select Hyperion, then Refresh.

Deleting a POV

You can delete a POV that has been saved in a workbook.

- To delete a POV:
- 1 Click POV Manager or select Hyperion, then POV Manager.
- 2 Expand the POV list.
- 3 From the POV drop-down list, select the worksheet that contains the POV that you want to delete.
- 4 Select the POV that you want to delete.
- 5 Click Delete.
- 6 Click Close.
- 7 To refresh the worksheet, select Hyperion, then Refresh. For Office 2007, click Refresh in the Review section of the **Hyperion** ribbon.

Printing POV Members in Header and Footer

You can print active POV members of the POV toolbar in the header or footer of an Excel document.

- To print POV members in the header or footer:
- 1 In Excel, select File, then Page Setup.
- 2 Under Header/Footer, click Custom Header or Custom Footer.
- 3 In the Left section, Center section, or Right section box, type a statement that includes POV: {}, such as My current POV members are POV:{}.
- 4 Click OK.
- 5 In Page Setup, click OK.

Using the Migration Utility

In This Chapter

| Considerations Before Converting Workbooks | 227 |
|--|-----|
| Converting One Workbook. | |
| Converting Multiple Workbooks | 228 |
| Migrating Connections for Functions | 229 |

The migration utility converts existing Financial Management and Hyperion Enterprise spreadsheet add-in functions to the new syntax when you upgrade. You can convert a single workbook or convert multiple workbooks using the batch option.

Considerations Before Converting Workbooks

You can convert workbooks that contain Financial Management Retrieve Data functions or Hyperion Enterprise HP Retrieve and VBA Retrieve functions from previous releases by using a migration utility. For example, you can convert Financial Management functions such as HFMVal, HFMLnk, HFMLab, HFMDes, and HFMCur and Hyperion Enterprise functions such as HPVal, HPLnk, HPCur, HPHea, HPCde, and HPFul.

The utility might not be able to convert all of your existing functions. Some functions might require manual adjustment. The following are considerations for converting workbooks.

For functions that use cell references, the following functions can be converted:

- If every parameter in the function is a cell reference, the function will convert. For example: =HFMVal(\$B\$1&\$C\$1&\$B\$2&\$C\$3&\$B\$5&\$C\$5&\$B\$6&\$C\$6).
- If the dimension parameters are specified in the function, the members are cell references, but the period separator is hard coded in the function, the function will convert. For example: =HFMVal("S#"&D2&".Y#"&D3&".VW#"&D5&".')

The following functions that use cell references are not converted:

- If the dimension parameters are specified in the function and the members and period separator are cell references, the function is not converted. For example: =HFMVal ("S#"&E2&"Y#"&E3&"VW#"&E5), where E2=Actual, E3=2004, E5="<Scenario View>."
- If the dimension parameters are specified in the function, the members are cell references, but the period separator is in a separate cell, the function is not converted. For example: =HFMVal("S#"&F2&C1&"Y#"&F3&C1&"VW#&F5&C1), where C1=. (period separator).

If the application specified in the function is a cell reference, the function does not convert properly.

Before you run the migration utility, ensure that the path is correct (the default path is C: \Hyperion\SmartView). During the migration process, Microsoft Excel inserts the original path of the add-in file to functions, which can make the functions too long and can cause problems in the migration process. Smart View functions cannot contain more than 256 characters per a limit within Excel.

Converting One Workbook

You can convert a workbook with existing Financial Management spreadsheet add-in functions to the new Hyperion Smart View syntax. For example, if you have a spreadsheet that contains the HFMVal(POV) function, it is converted to the HsGetValue(POV) function.

- To convert one workbook:
- 1 Select **Hyperion**, then **Functions**, and then **Migrate Active Workbook**.
- 2 If your functions contain application references, you must map the application to the corresponding connection.
- 3 Click Convert.
- 4 On the conversion completed message, click **OK**.

The system displays the migration results, including a list of any functions that failed to convert. You can manually adjust those functions.

- 5 To save the conversion results, click Save Result.
- 6 Select a location to store the results file, and click **Save**.
- 7 Click Close.

Converting Multiple Workbooks

You can convert multiple workbooks in a batch operation.

Note:

Financial Management users skip step 3 and Hyperion Enterprise users skip steps 6 and 7 in the following procedure.

- To convert multiple workbooks:
- 1 Select **Hyperion**, then **Functions**, and then **Migrate Batch**.
- 2 Click Add and select the workbooks that you want to convert.
- 3 To select workbooks for conversion, perform an action:

- Select Select All/None to select all workbooks.
- Select the check box next to each workbook to select specific workbooks.
- Click Next. If your functions contain application references, you must map the application to the connection.
- 5 Click Next.

The system displays the migration results, including a list of any functions that failed to convert. You can manually adjust those functions.

- 6 Click Save Result.
- 7 Select a location to store the results file and click **Save**.
- 8 Click Done.

For Hyperion Enterprise users, the converted workbooks are automatically saved in the location of the original workbooks.

Migrating Connections for Functions

In Financial Management, you can select a connection or connection reference for functions that do not contain an application reference when you migrate to Smart View.

- To migrate connections for functions:
- 1 Select Hyperion, then Function Builder, and then Migrate Active Connections (HFM).
- 2 From Function Migration Application reference, select an option:
 - Do not update functions with a connection reference. This migrates the existing functions without modification.
 - Add connection name to existing functions, then select a connection name from the Connection Name list. This updates all functions with the specified connection name.
 - Update functions with reference to connection list within selected worksheet, then in Cell Reference, enter the cell to reference, for example, A2. This updates all functions with a cell reference in the current worksheet.
 - Update functions with reference to connection list on a new worksheet, then enter the Worksheet name, and Cell Reference. This updates all functions with a cell reference to a different worksheet in the workbook.

Tip:

For the Update functions options, you can create a drop-down list in any cell on the worksheet to be used as a reference. Select Hyperion, then Functions, and then Insert Connection List to display a list of connections from which to choose in the current cell. This cell may be used as reference within functions to refer to a connection name.

3 Click OK.

VBA Functions

In This Chapter

| Using VBA Functions for Smart View | . 231 |
|--|-------|
| About Visual Basic Menu Equivalent Functions | .323 |
| Error Codes | .330 |

Using VBA Functions for Smart View

Smart View enables you to customize and automate common tasks by using Visual Basic for Applications (VBA) functions.

The menu equivalents are Visual Basic functions that execute the Hyperion menu commands.

Note:

To use VBA functions, the sheet must be active.

Migrating Legacy VBA Applications

To work in Smart View, VBA applications created using VBA functions from Oracle's Hyperion® Essbase® Spreadsheet Toolkit must be migrated to Smart View, although you can continue to use them in Oracle Essbase Spreadsheet Add-in.. In most cases, you can replace the prefix "EssV" with "Hyp" for analogous function names in your VBA applications. See "VBA Functions" on page 238 for a list of supported VBA functions. For example, for EssVRemoveOnly, change the name to HypRemoveOnly. For VBA menu functions, replace "EssMenuV" with "HypMenuV." For example, for EssMenuVZoomIn, change the name to HypMenuVZoomIn. Additionally, you must replace the declarations from essxlvba.txt with the declarations in smartview.bas.

Creating a Visual Basic Module

To use the VBA functions, you must first create Visual Basic modules to contain the VBA code.

- To create a Visual Basic module:
- 1 In Excel, select Tools, then Macro, then Visual Basic Editor to open the Visual Basic application.
- In the Visual Basic application, select **Insert**, then **Module**.

A new module is displayed with a labeled Module1 (or labeled with the next highest module number, if you have already inserted one or more modules).

Using Smart View VBA Functions

This procedure is an example of how to use VBA functions in Smart View.

- To use VBA functions (HypConnect in this example):
- 1 Select View, then Toolbars, and then Forms to display the Forms toolbar.
- 2 Select Tools , then Macro, then Visual Basic Editor to display the Visual Basic Editor.
- 3 In the Visual Basic Editor, select **File**, then **Import File**.
- 4 In Import File, select smartview.bas in \SmartView\bin to declare all functions or just the function you plan to use.

Tip:

For convenience, copy the entire text of smartview. bas into a separate module from the other VBA code.

- 5 Select **Insert**, then **Module** to create a module.
- 6 In the module, type the VBA code for the function.

For example, type the following VBA code for the HypConnect function:

```
Sub Conn()
  X=HypConnect(vtSheetName, User, Password, vtFriendlyName
End Sub
```

Substitute your own user name, password, and connection name for the data source provider.

- 7 From the Excel Forms toolbar, select the button to create a button. The Assign Macro dialog box is displayed.
- 8 In **Assign Macro**, select the name of the subroutine from the list of functions.

In this example, select Conn.

9 Click OK.

The function is now associated with the button that you just created.

- 10 Optional: Rename the button.
- 11 To run this function, select **Tools**, then **Macro**, and then **Macros**, select the function name, then click

Alternatively, you can click the button that you just created.

Declaring Functions

Before you can use Visual Basic functions in an Excel project, you must declare them in a module. A declaration outlines the necessary elements of the function so Visual Basic can run it. You must declare only those functions you plan to use, or you may declare all the VBA functions.

- To declare all Smart View VBA functions:
- 1 In Visual Basic Editor, select File, then Import File.
- From Import File, select smartview.bas in \SmartView\bin.

The file is copied into the module. Now you can use any Smart View Excel VBA function in your program. If there are functions you do not need in your module, you can delete their declarations.

- To declare individual Smart View VBA functions:
- 1 In Visual Basic Editor, select Insert, then File.
- Move the cursor to the top of the module.
- 3 Type the appropriate declarations for the functions you will use.

Refer to the description of each function for its declaration, or refer to smartview. bas. For example:

Declare Function HypConnect Lib "HsAddin.dll" (ByVal vtSheetName As Variant, ByVal vtUserName As Variant, ByVal vtPassword As Variant, ByVal vtFriendlyName As Variant) As Long

Guidelines for Declaring Functions

When typing the declaration, observe the following guidelines:

- Do not substitute specific values for each parameter name.
 - In the HypConnect example, the first parameter is vtSheetName. In the declaration, you type it as vtSheetName. When you actually call the HypConnect function in your VBA module, you substitute the appropriate name of the sheet.
- Type the declaration on one line.

Calling Functions

Once the functions are declared at the top of the module, you can call them in your VBA code. When you call a function, you tell it to perform its intended action and return a value. You can then test the returned value to see if the function ran successfully.

The following example shows the syntax for HypConnect.

HypConnect(vtSheetName, vtUser, vtPassword, vtFriendlyName)

When you call a function, observe the following guidelines:

- Substitute the appropriate value for each parameter, shown in italics.
- Type a value for every parameter. All parameters are required.
 - Many parameters have default values. If you do not want to specify a value for such parameters, type Null or Empty. This tells Essbase to use the default value for that parameter.
- Assign the function to a variable. After the function runs, the variable stores the return value, which indicates the success or failure of the function.

Dynamic Link Views

A link view is used to display the details about a data point in an adjacent window without disturbing the contents in the main window. Link views can be either static or dynamic.

In a static link view, the link action is predefined and details about a data point being queried are displayed in the adjacent window. Static link view behavior is already built in to Smart View.

With a dynamic link view, VBA programmers have the option to change the link behavior as required. Using the set options, you can change the row, column, POV, column information, and the connection information.

For instructions on working with dynamic link views, see:

- "Setting Up Dynamic Link Views" on page 235
- "Automating Macro Execution" on page 235

The VBA functions related to dynamic link view:

- HypUseLinkMacro
- HypSetLinkMacro
- HypGetLinkMacro
- HypGetSourceGrid

Note:

For all of the following dynamic link view VBA functions, it is assumed that a call has already been made to HypGetSourceGrid to initialize the dynamic link query, which contains the information about the active data source and the grid on the sheet.

- HypGetConnectionInfo
- HypSetConnectionInfo
- HypGetRowCount
- HypGetColCount
- HypGetPOVCount
- HypGetRowItems
- HypGetColItems
- HypGetPOVItems
- HypSetRowItems
- HypSetColItems
- HypSetPOVItems

When the dynamic link query has been initialized, all the subsequent setinfo, getinfo, displaytolinkview calls are performed on that saved dynamic link query. If the user changes the grid on the sheet and wants to perform the dynamic link action as per the new grid, the user must again initialize the query, using the various setinfo calls available.

Prerequisite to HypGetSourceGrid is that a connected grid must exist on the active sheet and a valid data point should be selected.

Setting Up Dynamic Link Views

Use dynamic link views to customize the link behavior according to your requirements. With dynamic link view, you can change the row, column, POV, and column information as well as the connection information.

- To set up a dynamic link view:
- 1 Set the HypUseLinkMacro flag to true.
- 2 Set the macro name to run.

The macro name you set should contain all the function calls to initialize the grid and to set the connection, row, POV, and column items as needed.

- 3 Connect the sheet and retrieve the appropriate grid onto the sheet.
- 4 Select a data point on the sheet.
- 5 Select Hyperion, then Link View, then Visualize in Excel. For Office 2007, click Visualize in Excel in the Visualize section of the Hyperion ribbon.

The macro name set in step 2 is executed and the link action is performed.

Note:

When the HypUseLinkMacro flag is set to false, the predefined link query is performed.

Automating Macro Execution

You can automate execution of a macro through the Hyperion menu.

- To set up a macro to execute manually through the Hyperion menu:
- 1 Set the HypUseLinkMacro flag to false.
- 2 Connect the sheet and retrieve the grid you want onto the sheet.
- 3 Select a data point on the sheet.
- Run the macro which contains all the function calls to initialize the grid and set the connection, row, column, and POV items.

VBA Parameters

Most Visual Basic functions require you to supply one or more parameters. Table 13 lists the parameter types and how to supply a valid value for each type:

Table 13 VBA Parameters

| Parameters | Valid Values |
|-----------------|---|
| Text | A word or phrase or name in quotes. For example, |
| | "Smart View" |
| | "[Book2.xls]Sheet1" |
| Boolean | True |
| | False |
| Range Object | A cell, row or column, one or more selections of cells, or a three-dimensional range address, surrounded by quotes. For example: |
| | RANGE("A1") |
| | RANGE("A1:B2") |
| | RANGE("A1:B2") |
| | RANGE("G:G,I:I,K:K") |
| | RANGE("A1:B5,C1:C10,D5:L8") |
| | RANGE("Sheet1!C3:R20,Sheet2!C3:R20") |
| Number | A number without quotes and without commas. For example: |
| | 1 |
| | 2.5 |
| | 50000 |
| List of Strings | A list of Text values, separated by commas. For example: |
| | "Qtr1", "Actual", "Oregon" |
| Constant | A predefined constant from smartview.bas. |
| Default Value | Null |
| | Empty |
| | Note: Many parameters have default values or behavior that the function uses if you specify Null or Empty. If you do not specify a value for such parameters, use Null or Empty. See the description of each function for default values of such parameters. |

VBA Return Values

The Smart View VBA functions return values indicating the success or failure of the function. The functions return several types of values:

0 (zero) — the function was successful. Functions can run successfully but still have undesirable results.

- 1 typically means the user pressed Escape or clicked Cancel from a dialog box.
- -1 indicates a valid return value, True.
- Negative number the function failed due to a problem with the client machine, a problem with the syntax, or a problem with the local environment.
- large positive number, the function failed due to a problem originating on the server, such as the server not running or an invalid user name.

VBA Local Return Values

Local problems are represented by negative return values:

- 0 Function ran successfully
- -1 Valid return value, True
- -2 Termination error
- -3 Initialization error
- -4 Spreadsheet is not yet connected to the server
- -6 Not used
- -7 Spreadsheet has become unstable
- -8 No Undo information exists
- -9 Operation has been canceled
- -12 Undo is not enabled
- -13 Not enough memory resources are available
- -14 Appropriate dialog box could not be displayed
- -15 Function contains an invalid parameter
- -16 Calculation is in progress
- -17 Obsolete setting
- -18 Operation is not allowed because the spreadsheet is in formula preservation mode
- -19 Operation cannot take place on the specified sheet
- -20 Current sheet cannot be determined
- -21 Spreadsheet name was not specified and no active sheet is selected
- -22 Calculation cannot be canceled because no calculation is running
- -23 Selection parameter is invalid
- -25 Cascade list file cannot be created, or you are attempting to cascade while the spreadsheet is embedded in another document
- -26 Spreadsheet macros cannot be run due to a licensing agreement
- -27 Spreadsheet macros which update the database cannot be run due to a licensing constraint

- -28 Database cannot be updated because you have a read-only license for the database
- -29 Obsolete setting
- -30 Menu is removed already
- -31 Menu is added already
- -39 The specified worksheet is protected. Unprotect the worksheet and try the operation again.
- -40 Calc script not found

VBA Functions

Table 14 lists the Smart View VBA functions alphabetically and specifies the data source provider that support the function. A detailed description for each function, including the syntax, parameter, return value, and sample code, follows the table.

Note:

For this release, VBA functions for Planning are not supported. To use ad hoc with data forms, you must be connected to an Essbase or Hyperion Enterprise data source.

Table 14 VBA Functions and Supported Providers

| VBA Functions | Applicable Data Sources |
|---------------------------|---|
| HypCalculate | Financial Management, Hyperion Enterprise |
| HypCalculateContribution | Financial Management, Hyperion Enterprise |
| HypCell | Essbase |
| HypConnect | Essbase, Financial Management , Hyperion Enterprise |
| HypConnected | Essbase, Financial Management, Hyperion Enterprise |
| HypConnectionExists | Essbase, Financial Management, Hyperion Enterprise |
| HypConnectToAPS | Essbase, Planning, Oracle BI EE |
| HypConsolidate | Financial Management, Hyperion Enterprise |
| HypConsolidateAll | Financial Management, Hyperion Enterprise |
| HypConsolidateAllWithData | Financial Management, Hyperion Enterprise |
| HypCreateConnection | Essbase, Financial Management, Hyperion Enterprise |
| HypCreateConnectionEX | Essbase, Planning, Oracle BI EE, Financial Management, Hyperion Enterprise |
| HypDeleteCalc | Essbase |

| VBA Functions | Applicable Data Sources |
|-------------------------------|--|
| HypDisconnect | Essbase, Financial Management, Hyperion Enterprise |
| HypDisconnectEx | EssbasePlanningOracle BI EE, Financial Management |
| HypDisconnectFromAPS | Essbase, Planning, Oracle BI EE |
| HypExecuteCalcScript | Essbase |
| HypExecuteMDXEx | Essbase |
| HypExecuteQuery | Essbase |
| HypFindMember | Essbase, Financial Management, Hyperion Enterprise |
| HypForceCalculate | Financial Management, Hyperion Enterprise |
| HypForceCalculateContribution | Financial Management, Hyperion Enterprise |
| HypForceTranslate | Financial Management, Hyperion Enterprise |
| HypFreeDataPoint | Essbase, Financial Management, Hyperion Enterprise |
| HypGetAncestor | Essbase |
| HypGetChildren | Essbase |
| HypGetColCount | Essbase, Financial Management, Hyperion Enterprise |
| HypGetColltems | Essbase, Financial Management, Hyperion Enterprise |
| HypGetConnectionInfo | Essbase, Financial Management, Hyperion Enterprise |
| HypGetCurrentAPSURL | Essbase, Planning, Oracle BI EE |
| HypGetDataPoint | Essbase, Financial Management, Hyperion Enterprise |
| HypGetGlobalOption | Essbase, Financial Management, Hyperion Enterprise |
| HypGetLinkMacro | Essbase, Financial Management, Hyperion Enterprise |
| HypGetOverrideFlag | Essbase, Planning, Oracle BI EE |
| HypGetParent | Essbase |
| HypGetPOVCount | Essbase |
| HypGetPOVItems | Essbase, Financial Management |
| HypGetRowCount | Essbase, Financial Management |
| HypGetRowItems | Essbase, Financial Management |
| HypGetSheetOption | Essbase, Financial Management |
| HypGetSourceGrid | Essbase, Financial Management |

| VBA Functions | Applicable Data Sources |
|-------------------------------------|---|
| HypGetSubstitutionVariable | Essbase |
| HypIsAttribute | Essbase |
| HypIsConnectedToAPS | Essbase, Planning, Oracle BI EE |
| HypIsDescendant | Essbase |
| HypIsExpense | Essbase |
| HypIsParent | Essbase |
| HypisUDA | Essbase |
| HypKeepOnly | Essbase, Financial Management, Hyperion Enterprise |
| HypListCalcScripts | Essbase |
| HypMigrateConnectionToDataSourceMgr | Essbase |
| HypOtlGetMemberInfo | Essbase |
| HypPivot | Essbase, Financial Management, Hyperion Enterprise |
| HypPivotToGrid | Essbase, Financial Management |
| HypPivotToPOV | Essbase, Financial Management |
| HypQueryMembers | Essbase |
| HypRedo | Essbase, Financial Management, Hyperion Enterprise |
| HypRemoveConnection | Essbase, Financial Management, Hyperion Enterprise |
| HypRemoveOnly | Essbase, Financial Management, Hyperion Enterprise |
| HypResetFriendlyName | Essbase, PlanningOracle BI EE, Financial Management |
| HypRetrieve | Essbase, Financial Management, Hyperion Enterprise |
| HypRetrieveRange | Essbase |
| HypSetActiveConnection | Essbase, Financial Management, Hyperion Enterprise |
| HypSetBackgroundPOV | Essbase, Financial Management, Hyperion Enterprise |
| HypSetCellsDirty | Essbase, Financial Management, Hyperion Enterprise |
| HypSetColItems | Essbase, Financial Management, Hyperion Enterprise |
| HypSetConnAliasTable | Essbase |
| HypSetConnectionInfo | Essbase, Financial Management, Hyperion Enterprise |
| HypSetCurrentUserAPSURL | Essbase, Planning, Oracle BI EE |

| VBA Functions | Applicable Data Sources |
|----------------------------|--|
| HypSetGlobalOption | Essbase, Financial Management, Hyperion Enterprise |
| HypSetLinkMacro | Essbase, Financial Management, Hyperion Enterprise |
| HypSetMenu | Essbase, Financial Management, Hyperion Enterprise |
| HypSetOverrideFlag | Essbase, Planning, Oracle Business Intelligence Enterprise Edition |
| HypSetPOV | Essbase, Financial Management, Hyperion Enterprise |
| HypSetPOVItems | Essbase, Financial Management, Hyperion Enterprise |
| HypSetRowItems | Essbase, Financial Management, Hyperion Enterprise |
| HypSetSheetOption | Essbase, Financial Management, Hyperion Enterprise |
| HypSetSubstitutionVariable | Essbase |
| HypSubmitData | Essbase, Financial Management, Hyperion Enterprise |
| HypTranslate | Financial Management, Hyperion Enterprise |
| HypUndo | Essbase, Financial Management, Hyperion Enterprise |
| HypUseLinkMacro | Essbase, Financial Management, Hyperion Enterprise |
| HypZoomIn | Essbase, Financial Management, Hyperion Enterprise |
| HypZoomOut | Essbase, Financial Management, Hyperion Enterprise |

HypCalculate

Description

HypCalculate() calls the Calculate method for Financial Management data sources.

Syntax

HypCalculate (vtSheetName, vtRange) ByVal vtSheetName As Variant By Val vtRange As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

Example

```
Declare Function HypCalculate Lib "HsAddin" (ByVal vtSheetName As Variant, ByVal vtRange As Variant) As Long

sts = HypCalculate ("Sheet1", Empty)
```

HypCalculateContribution

Description

HypCalculateContribution() calls the Calculate Contribution method for Financial Management data sources.

Syntax

HypCalculateContribution (vtSheetName, vtRange) ByVal vtSheetName As Variant By Val vtRange As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls]Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

Example

```
Declare Function HypCalculateContribution Lib "HsAddin" (ByVal vtSheetName As Variant, ByVal vtRange As Variant) As Long

sts = HypCalculateContribution ("Sheet1", Empty)
```

HypCell

Description

HypCell() retrieves a cell value for a single member combination.

Syntax

```
HypCell(vtSheetName, ParamArray MemberList())
ByVal vtSheetName As Variant
ByVal ParamArray MemberList() As Variant
```

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

MemberList: A list of strings which describe the member combination for which a data value will be retrieved. If MemberList is Null or Empty, the top level value is used. Represent members as "Dimension#Member"; for example, "Year#Jan" or "Market#East".

Return Value

Returns the value of the data point if successful. Returns #No Connection if the sheet cannot be determined or is not connected to a data source. Returns "Invalid Member MemberName or dimension DimensionName" if a member is incorrect.

Example

```
Declare Function HypCell Lib "HsAddin" (ByVal vtSheetName As Variant,
ParamArray MemberList() As Variant) As Variant
Sub InCell()
Dim X As String
X=HypCell("[Book2.xls]Sheet1", "Year#Qtr1", "Scenario#Actual",
"Market#Oregon")
    If X = "#No Connection" Then
       MsgBox("Not logged in, or sheet not active.")
       If Left(X, 15) = "#Invalid member" then
          MsgBox("Member name incorrect.")
          MsgBox(X + " Value retrieved successfully.")
       End If
    End If
End Sub
```

Note:

The value of the data point returned is not placed in a cell in the spreadsheet automatically. To place the value in a cell, use the Visual Basic select method and the ActiveCell property. See your Visual Basic documentation for more information.

HypConnect

Description

HypConnect() logs into a data source provider and associates the worksheet with that connection. HypConnect() must be called for each sheet in order to associate this connection with that sheet.

Syntax

HypConnect(vtSheetName, vtUserName, vtPassword, vtFriendlyName)

ByVal vtSheetName As Variant

ByVal vtUserName As Variant

ByVal vtPassword As Variant

ByVal vtFriendlyName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtUserName: Text name of a valid user for the data source provider.

vtPassword: Text name of the password for this user.

vtFriendlyName: The friendly connection name for the data source provider. This is the connection name created by HypCreateConnection.

Return Value

Returns 0 if successful. A negative number indicates a local failure. A return value greater than zero indicates a failure on the server.

Example

Declare Function HypConnect Lib "HsAddin.dll" (ByVal vtSheetName As Variant, ByVal vtUserName As Variant, ByVal vtPassword As Variant, ByVal vtFriendlyName As Variant) As Long

Sub Conn()

```
X=HypConnect(Empty, username, password, "My Sample Basic")
End Sub
```

HypConnected

Description

HypConnected() provides the connection status of the sheet. A true value indicates that the sheet is connected to a provider; a false value indicates that the sheet is not connected.

Syntax

HypConnected (vtSheetName)

ByVal vtSheetName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

Return Value

If the sheet is connected, return value of the variant is -1. If the sheet is not connected, return value of the variant is 0. If the sheets contains Report Designer objects, return value is 2.

Example

```
Declare Function HypConnected Lib "HsAddin" (ByVal vtSheetName As Variant)
As Variant
Sub Sample_HypConnected
  Dim X as Variant
   X = HypConnected("Sheet1")
End sub
```

HypConnectionExists

Description

HypConnectionExists() is used to check if a particular connection name exists in the list of all connections as viewed in the Data Source Manager. The particular connection may or may not be active (i.e., connected).

Syntax

HypConnectionExists(vtConnectionName)

ByVal vtConnectionName as Variant

Parameters

vtConnectionName: Name of the connection to search for in the list of all connections. It is not case-sensitive.

Return Value

Boolean. If successful, return value is TRUE; otherwise, return value is FALSE.

Example

```
Declare Function HypConnectionExists Lib "HsAddin.dll" (ByVal
vtConnectionName As Variant) As Variant
Sub Sample_SetActiveConnection
   Dim bIsConnection as Boolean
   bIsConnection = HypConnectionExists ("Demo_Basic")
End sub
```

HypConnectToAPS

Description

HypConnectToAPS connects Smart View to Provider Services with the URL set in Hyperion Options. By default it connects to the default Provider Services URL, but if the user has set it to be overridden by a specific Provider Services URL, this specific Provider Services URL will be used to create and connect to the connection.

Syntax

HypConnectToAPS()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypConnectToAPS Lib "HsAddin.dll" () As Long
Sub SubHypConnectToAPSTest()
        Dim 1Ret As Long
```

```
1Ret = HypConnectToAPS()
End Sub
```

HypConsolidate

Description

HypConsolidate calls the Consolidate method for Financial Management data sources.

Syntax

HypConsolidate (vtSheetName, vtRange)

ByVal vtSheetName As Variant

By Val vtRange As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

Example

```
Declare Function HypConsolidate Lib "HsAddin" (ByVal vtSheetName As
Variant, ByVal vtRange As Variant) As Long
sts = HypConsolidate ("Sheet1", Empty)
```

HypConsolidateAll

Description

HypConsolidateAll() calls the Consolidate All method for Financial Management data sources.

Syntax

HypConsolidateAll (vtSheetName, vtRange)

ByVal vtSheetName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

Example

```
Declare Function HypConsolidateAll Lib "HsAddin" (ByVal vtSheetName As
Variant, ByVal vtRange As Variant) As Long
sts = HypConsolidateAll ("Sheet1", Empty)
```

HypConsolidateAllWithData

Description

HypConsolidateAllWithData calls the Consolidate All With Data method for Financial Management data sources.

Syntax

HypConsolidateAllWithData (vtSheetName, vtRange)

ByVal vtSheetName As Variant

By Val vtRange As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

Example

Declare Function HypConsolidateAllWithData Lib "HsAddin" (ByVal vtSheetName As Variant, ByVal vtRange As Variant) As Long

```
sts = HypConsolidateAllWithData ("Sheet1", Empty)
```

HypCreateConnection

Description

HypCreateConnection() creates a connection to the data source provider from the specified information. See also HypCreateConnectionEX.

Note:

Planning users who want to add data sources in the Data Source Manager must use HypCreateConnectionEX.

Note:

Use HypConnect to establish the connection.

Syntax

HypCreateConnection(vtUserName, vtPassword, vtProvider, vtProviderURL, vtServerName, vtApplicationName, vtDatabaseName, vtFriendlyName, vtDescription)

ByVal vtSheetName As Variant — not used

ByVal vtUserName As Variant

ByVal vtPassword As Variant

ByVal vtProvider As Variant

ByVal vtProviderURL As Variant

ByVal vtServerName As Variant

ByVal vtApplicationName As Variant

ByVal vtDatabaseName As Variant

ByVal vtFriendlyName As Variant

ByVal vtDescription As Variant

Parameters

vtUserName:Text name of a valid user on the server.

vtPassword: Text name of the password for this user.

vtProvider: Description for the data source provider. Supported vtProvider types:

- New: Global Const HYP_ESSBASE = "Essbase"
- New: Global Const HYP_PLANNING = "Planning"
- New: Global Const HYP_OBIEE = "OBIEE"
- **Deprecated:** Global Const HYP_ANALYTIC_SERVICES = "Analytic Provider Services"
- Global Const HYP_FINANCIAL_MANAGEMENT = "Hyperion Financial Management"

vtProviderURL: Data source provider URL which to connect.

vtServerName: Name of the server on which the application resides.

vtApplication: Name of the application.

vtDatabase: Name of the database.

vtFriendlyName: Connection name for the data source provider.

vtDescription: Description for the data source provider.

Return Value

Returns 0 if successful. A negative number indicates a local failure. A return value greater than zero indicates a failure on the server.

Example

Declare Function HypCreateConnection Lib "HsAddin" (ByVal vtSheetName As Variant, ByVal vtUserName As Variant, ByVal vtPassword As Variant, ByVal vtProvider As Variant, ByVal vtProviderURL As Variant, ByVal vtServerName As Variant, ByVal vtApplicationName As Variant, ByVal vtDatabase As Variant, ByVal vtFriendlyName As Variant, ByVal vtDescription As Variant) As Long

```
Sub Conn()
    X=HypCreateConnection(Empty, username, password, HYP_ANALYTIC_SERVICES,
    "http://localhost:13080/smartview/SmartView", "localhost", "Sample",
    "Basic", "My Connection", "Analytic Provider Services")
End Sub
```

HypCreateConnectionEX

Description

HypCreateConnectionEX is a superset of HypCreateConnection; it has additional parameters that enable use of the Data Source Manager. Planning users who want to add data sources in the Data Source Manager must use HypCreateConnectionEX.

Syntax

HypCreateConnection(vtUserName, vtPassword, vtProvider, vtProviderURL, vtServerName, vtApplicationName, vtDatabaseName, vtFriendlyName, vtDescription)

ByVal vtProviderType As Variant

ByVal vtServerName As Variant

ByVal vtApplicationName As Variant

ByVal vtDatabaseName As Variant

ByVal vtFormName As Variant

ByVal vtProviderURL As Variant

ByVal vtFriendlyName As Variant

ByVal vtUserName As Variant

ByVal vtPassword As Variant

ByVal vtDescription As Variant

ByVal vtReserved1 As Variant (reserved for future use)

ByVal vtReserved2 As Variant (reserved for future use)

Parameters

vtProviderType: Description for the data source provider. Supported vtProvider types:

- Global Const HYP_ESSBASE = "Essbase"
- Global Const HYP_PLANNING = "Planning"
- Global Const HYP_OBIEE = "OBIEE"
- Global Const HYP_FINANCIAL_MANAGEMENT = "Hyperion Financial Management"

vtServerName: Name of the server on which the application resides.

vtApplication: Name of the application.

vtDatabase: Name of the database.

vtFormName: Name of the data form. Required to create Oracle Hyperion Planning, Fusion Edition connection in Data Source Manager under Favorites

vtUserName:Text name of a valid user on the server.

vtPassword: Text name of the password for this user.

vtProviderURL: Data source provider URL which to connect.

vtFriendlyName: Connection name for the data source provider.

vtDescription: Description for the data source provider.

Return Value

Returns 0 if successful. A negative number indicates a local failure. A return value greater than zero indicates a failure on the server.

Example

Public Declare Function HypCreateConnectionEx Lib "HsAddin" (ByVal vtProviderType As Variant, ByVal vtServerName As Variant, ByVal vtApplicationName As Variant, ByVal vtDatabaseName As Variant, ByVal vtFormName As Variant, ByVal vtProviderURL As Variant, ByVal vtFriendlyName As Variant, ByVal vtUserName As Variant, ByVal vtPassword As Variant, ByVal vtDescription As Variant, ByVal vtReserved1 As Variant, ByVal vtReserved2 As Variant) As Long

```
Sub CreateConnExTest()
Dim lRet As Long
1Ret = HypCreateConnectionEx("Essbase", "server12", "Demo", "Basic", "",
"", "My Demo", "system", "password", "", "", "")
lRet = HypCreateConnectionEx("Planning", "planqe14", "TotPlan", "", "/Forms/
Smart View Forms/01 Product Revenue", "", "My Planning VBA Conn", "admin",
"password", "", "", "")
End Sub
```

HypDeleteCalc

Description

HypDeleteCalc() allows the user to delete a calculation script object from an Analytic Server.

Syntax

HypDeleteCalc (vtSheetName, vtApplicationName, vtDatabaseName, vtCalcScript)

ByVal vtSheetName As Variant

ByVal vtApplicationName As Variant

ByVal vtDatabaseName As Variant

vtSheetName: Text name of worksheet on which to operate. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtApplicationName: Specify the application name containing the calculation script.

vtDatabaseName: Specify the database name containing the calculation script.

vtCalcScript: Specify the calculation script name to be deleted.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypDeleteCalc Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtApplicationName As Variant, ByVal vtDatabaseName As
Variant, ByVal vtCalcScript As Variant) As Long
Sub Sample HypDeleteCalc
Dim X as Long
  X = HypDeleteCalc ("Sheet1", "Sample", "Basic", "CalcYear")
End Sub
```

HypDisconnect

Description

HypDisconnect() logs out from the data source provider.

Syntax

```
HypDisconnect(vtSheetName, bLogoutUser)
ByVal vtSheetName As Variant
ByVal bLogoutUser As Boolean
```

Parameters

vtSheetName: Text name of worksheet on which to operate. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

bLogoutUser: Optional. Set to True to disconnect and log out from the provider session. Default value is False.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypDisconnect Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal bLogoutUser As Boolean) As Long
Sub DisConn()
   X=HypDisconnect(Empty, True)
End Sub
```

HypDisconnectEx

Description

HypDisconnectEx disconnects the connection with the the connection (friendly) name passed in the argument. The connection to be disconnected need not be associated as in HypDisconnect.

Syntax

HypDisconnectEx (vtConnFriendlyName) ByVal vtConnFriendlyName as Variant

Parameters

vtConnFriendlyName: the friendly connection name

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

```
Declare Function HypDisconnectEx Lib "HsAddin.dll" (ByVal
vtConnFriendlyName As Variant) As Long
Sub SubDisconnectExTest()
       Dim 1Ret As Long
        lRet = HypDisconnectEx("My Sample")
End Sub
```

HypDisconnectFromAPS

Description

HypDisconnectFromAPS disconnects Smart View from Provider Services. It first disconnects from all connected datasources (connections in the Data Source Manager) and then disconnects from the main Provider Services.

Syntax

HypDisconnectFromAPS()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypDisconnectFromAPS Lib "HsAddin.dll" () As Long
Sub DisconnectAPSTest()
        Dim 1Ret As Long
        1Ret = HypDisconnectFromAPS()
End Sub
```

HypDisplayToLinkView

Description

Syntax

```
HypDisplayToLinkView (vtDocumentType, vtDocumentPath)
ByVal vtDocumentType As Variant
vtDocumentPath As Variant
```

Parameters

```
vtDocumentType
vtDocumentPath
```

Return Value

Examples

Declare Function HypDisplayToLinkView Lib "HsAddin.dll" (ByVal vtDocumentType As Variant, ByVal vtDocumentPath As Variant) As Long

HypDisplayTwoDimArray

Description

Syntax

HypDisplayTwoDimArray (vtMemberList)

ByRef vtMemberList() As Variant

Parameters

vtMemberList:

Return Value

Example

Declare Function HypDisplayTwoDimArray Lib "HsAddin.dll" (ByRef vtMemberList () As Variant) As Long

HypExecuteCalcScript

Description

HypExecuteCalcScript() uses a calculation script (business rule script) to initiate a calculation on the server.

Syntax

HypExecuteCalcScript (vtSheetName, vtCalcScript, bSynchronous)

ByVal vtSheetName As Variant

ByVal vtCalcScript As Variant

ByVal bSynchronous As Boolean

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls]Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtCalcScript: Text name of the calculation script on the Analytic Server in the database directory to run. To run the default calculation script, use "Default".

bSynchronous: Boolean value indicating whether the calculation script should be run synchronously. If synchronous is Null or Empty, True is used. Currently this flag is unused.

Return Value

Returns 0 if successful. A negative number indicates a local failure. A return value greater than zero indicates a failure in one of the servers.

Example

```
Declare Function HypExecuteCalcScript Lib "HsAddin.dll" (ByVal vtSheetName
As Variant, ByVal vtCalcScript As Variant, ByVal bSynchronous As Variant)
As Long
Sub RunCalculate()
X = HypExecuteCalcScript (Empty, "Default", False)
   If X = 0 Then
      MsgBox("Calculation complete.")
   Else
      MsgBox("Calculation failed.")
   End If
End Sub
```

HypExecuteMDXEx

Description

HypExecuteMDXEx() executes an MDX query whose results are output in a data structure but are not displayed on the worksheet. (If you want to display the query results on a worksheet, use HypExecuteQuery instead.)

Syntax

```
HypExecuteMDXEx
ByVal vtSheetName As Variant,
ByVal vtQuery As Variant,
ByVal vtBoolHideData As Variant,
ByVal vtBoolDataLess As Variant,
ByVal vtBoolNeedStatus As Variant,
ByVal vtMbrIDType As Variant,
ByVal vtAliasTable As Variant,
```

```
ByRef outResult As MDX_AXES_NATIVE
) As Long
```

vtSheetName: Text name of worksheet to perform the action. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtQuery: The MDX query to be executed

vtBoolHideData: The Boolean flag hide or unhide data in the results

vtBoolDataless: The Boolean flag to get or avoid data in the results

vtBoolNeedStatus: The Boolean flag to get or avoid status info in the results

vtMbrIDType:

vtAliasTable: The Alias Table to be used

outResult: Pointer to a structure of type MDX_AXES. It contains the query output. (See the notes section for data types and support functions for this API)

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

vtBoolNeedStatus = True

```
Sub GetVBCompatibleMDXStructure (ByRef inStruct As MDX_AXES_NATIVE, ByRef
outStruct As MDX_AXES)
Public Declare Function HypExecuteMDXEx Lib "HsAddin.dll"
(ByVal vtSheetName As Variant, ByVal vtQuery As Variant, ByVal
vtBoolHideData As Variant, ByVal vtBoolDataLess As Variant, ByVal
vtBoolNeedStatus As Variant, ByVal vtMbrIDType As Variant, ByVal
vtAliasTable As Variant, ByRef outResult As MDX_AXES_NATIVE) As Long
Sub Example_HypExecuteMDXEx ()
Dim Query As Variant
Dim vtBoolHideData As Variant
Dim vtBoolDataLess As Variant
Dim vtBoolNeedStatus As Variant
Dim vtMbrIDType As Variant
Dim vtAliasTable As Variant
Dim result_Native As MDX_AXES_NATIVE ---- New Data Type (More Info in
Notes Section)
Dim result_VBCompatible As MDX_AXES ----New Data Type (More Info in
Notes Section)
Query = "select {Jan} on COLUMNS, {Profit} on ROWS from Sample.Basic"
vtBoolHideData = True
vtBoolDataLess = True
```

```
vtMbrIDType = "alias"
vtAliasTable = "none"
sts = HypConnect("Sheet1", "system", "password", "SB")
If sts = 0 Then
sts = HypExecuteMDXEx ("Sheet1", Query, vtBoolHideData, vtBoolDataLess,
vtBoolNeedStatus, vtMbrIDType, vtAliasTable, result_Native)
sts = GetVBCompatibleMDXStructure (result_Native, result_VBCompatible) ---
New support function ... More Info under Notes section
sts = HypDisconnect("Sheet1", True)
Else
End If
End Sub
```

Data Types Specific to HypExecuteMDXEx

The following data types apply exclusively to HypExecuteMDXEx:

MDX_CELL: The data type corresponding to a cell

MDX_PROPERTY: The data type containing properties info for members and dimensions

MDX_MEMBER: The data type for members information

MDX_DIMENSION: The data type for dimensions information

MDX_CLUSTER: The data type for cluster information

MDX_AXIS: The data type representing an axis

MDX_AXES: The root level structure containing a collection of axes and cells

MDX_AXES_NATIVE: The data type used as an out parameter for HypExecuteMDXEx. This structure should be converted to MDX_AXES using procedure GetVBCompatibleMDXStructure.

HypExecuteQuery

Description

HypExecuteQuery() executes an MDX query and displays the results on a worksheet. (If you do not want to display the query results on a worksheet, use HypExecuteMDXEx instead.)

Syntax

HypExecuteQuery (ByVal vtSheetName As Variant, ByVal vtMDXQuery As Variant) As Long ByVal vtSheetName As Variant

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtMDXQuery: The MDX query statement to be executed on the worksheet.

Return Value

Long. If successful, return value is 0; otherwise, returns the appropriate error code.

Example

```
Declare Function HypExecuteQuery Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtMDXQuery As Variant) As Long
Sub Sample_HypExecuteQuery ()
   Dim vtQuery As Varian
   vtQuery = "SELECT {([Jan])} on COLUMNS, {([East])} on ROWS from
                 Sample.Basic"
   sts = HypConnect ("Sheet1", "system", "password", "Sample_Basic")
   sts = HypExecuteQuery ("Sheet1", vtQuery)
   sts = HypDisconnect ("Sheet1", True)
End sub
```

HypFindMember

Description

HypFindMember() retrieves member information like dimension, alias, generation and level names.

Syntax

HypFindMember (vtSheetName, vtMemberName, vtAliasTable, vtDimensionName, vtAliasName, vtGenerationName, vtLevelName)

ByVal vtSheetName As Variant

ByVal vtMemberName As Variant

ByVal vtAliasTable As Variant

ByRef vtDimensionName As Variant

ByRef vtAliasName As Variant

ByRef vtGenerationName As Variant

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtMemberName: The name of the member. This parameter is required because there is no default value.

vtAliasTable: The name of the alias table to search for the alias name. If Null, the default alias table is searched.

vtDimensionName: The output parameter that contains the dimension, if successful.

vtAliasName: The output parameter that contains the alias name of the member, if successful.

vtGenerationName: The output parameter that contains the generation name of the member, if successful.

vtLevelName: The output parameter that contains the level name of the member, if successful.

Return Value

Returns zero if successful.

Example

Declare Function HypFindMember Lib "HsAddin.dll" (ByVal vtSheetName As Variant, ByVal vtMemberName As Variant, ByVal vtAliasTable As Variant, ByRef vtDimensionName as Variant, ByRef vtAliasName As Variant, ByRef vtGenerationName As Variant, ByRef vtLevelName As Variant) As Long

```
Sub FindMember()
  X = HypFindMember(Empty, "100", "Default", dimName, aliasName,
   genName, levelName)
  MsgBox (dimName)
  MsgBox (aliasName)
  MsgBox (genName)
  MsgBox (levelName)
End Sub
```

HypForceCalculate

Description

HypForceCalculate() calls the Force Calculate method for Financial Management data sources.

Syntax

HypForceCalculate(vtSheetName, vtRange)

ByVal vtSheetName As Variant

By Val vtRange As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

Example

```
Declare Function HypForceCalculate Lib "HsAddin" (ByVal vtSheetName As
Variant, ByVal vtRange As Variant) As Long
sts = HypForceCalculate ("Sheet1", Empty)
```

HypForceCalculateContribution

Description

HypForceCalculateContribution calls the Force Calculate Contribution method for Financial Management data sources.

Syntax

HypForceCalculateContribution (vtSheetName, vtRange) ByVal vtSheetName As Variant By Val vtRange As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

Example

```
Declare Function HypForceCalculateContribution Lib "HsAddin" (ByVal
vtSheetName As Variant, ByVal vtRange As Variant) As Long
sts = HypForceCalculateContribution (Empty, Empty)
```

HypForceTranslate

Description

HypForceTranslate calls the Force Translate method for Financial Management data sources.

Syntax

HypForceTranslate (vtSheetName, vtRange) ByVal vtSheetName As Variant By Val vtRange As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

```
Declare Function HypForceTranslate Lib "HsAddin" (ByVal vtSheetName As
Variant, ByVal vtRange As Variant) As Long
sts = HypForceTranslate (Empty, Empty)
```

HypFreeDataPoint

Description

HypFreeDataPoint() frees any memory allocated by HypGetDataPoint.

Syntax

HypFreeDataPoint()

ByRef vtInfo As Variant

Parameters

vtInfo: Variant array returned by HypGetDataPoint.

Return Value

Returns 0 if successful; returns -15 if not successful.

Example

See "HypGetDataPoint" on page 270 for an example of HypFreeDataPoint.

HypGetAncestor

Description

HypGetAncestor() returns the ancestor at any specific generation/level for the specified member.

Syntax

HypGetAncestor (vtSheetName, vtMemberName, vtLayerType, intLayerNum, vtAncestor)

ByVal vtSheetName As Variant

ByVal vtMemberName As Variant

ByVal vtLayerType As Variant

ByVal intLayerNum As Integer

ByRef vtAncestor As Variant

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls]Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtLayerType: Specify either "Gen" or "Level". If vtLayerType is Null or Empty, Gen is taken as default.

vtMemberName: Specify a member name. Required field.

intLayerNum: Specify the Level/Generation number. Required Field.

vtAncestor: Output. Contains the ancestor name on successful execution of the macro.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

Declare Function HypGetAncestor Lib "HsAddin.dll" (ByVal vtSheetName As Variant, ByVal vtMemberName As Variant, ByVal vtLayerType As Variant, ByVal intLayerNumber As Integer, ByRef vtAncestor As Variant) As Long

```
Sub Sample_HypGetAncestor
   Dim X as Long
   Dim vtAncestor as Variant
   X = HypGetAncestor ("Sheet1", "100-20", "Level", 1, vtAncestor)
End Sub
```

HypGetChildren

Description

HypGetChildren() returns the children for the specified member.

Syntax

HypGetChildren (vtSheetName, vtMemberName, intChildCount, vtChildArray)

ByVal vtSheetName As Variant

ByVal vtMemberName As Variant

ByVal intChildCount As Integer

ByRef vtChildArray As Variant

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtMemberName: Specify a member name. Required Field.

intChildCount: To restrict the number of children returned.

- ChildCount <= 0. All children are returned.
- ChildCount >0. The result set is limited to the number specified as the argument. If the result set is less than the specified argument, all result are returned.

vtChildArray: Output Result Vector that contains the list of the children. Its contents are unknown if the macro fails.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypGetChildren Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtMemberName As Variant, ByVal intChildCount As Integer,
ByRef vtChildNameArray As Variant) As Long
Sub Sample_HypGetChildren
   Dim vtChildren as Variant
   Dim vtChild as Variant
   Dim X as Long
      X = HypGetChildren ("sheet1", "Market", 0, vtChildren)
      If IsArray (vtChildren) Then
         For i = LBound (vtChildren) To UBound (vtChildren)
         VtChild = vtChildren (i)
     Next
   End If
End Sub
```

HypGetColCount

Description

HypGetColCount() returns the number of column dimensions.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypGetColCount()

Return Value

Returns the number of column dimensions if successful; otherwise, returns the negative error code.

Example

```
Declare Function HypGetColCount Lib "HsAddin.dll" () As Long
Sub Macro()
  Dim vtGrid as Variant
   Sts = HypConnect("Sheet1", "system", "password", "MyDemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetColCount ()
End sub
```

HypGetColItems

Description

HypGetColItems() returns the members present in the dynamic link query for the nth column dimensions.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

```
HypGetColItems(vtColumnID, vtDimensionName, vtMembers)
```

ByVal vtColumnID As Variant

ByRef vtDimensionName As Variant

ByRef vtMembers As Variant

Parameters

vtColumnID: The column number n.

vtDimensionName: Returns the nth column dimension name.

vtMembers: Returns members for the nth column dimensions.

Return Value

Returns 0 if successful; otherwise, returns the negative error code.

Example

```
Declare Function HypGetColItems Lib "HsAddin.dll" (ByVal vtColID As
Variant, ByRef vtDimensionName As Variant, ByRef vtMembernames As Variant)
As Long
Sub Macro()
   Dim vtGrid as Variant
   Dim vtDimensionName as Variant
  Dim vtMembers as Variant
  Sts = HypConnect("Sheet1", "system", "password", "AnamikaDemoBasic")
   Sts = HypRetrieve("Sheet1")
  Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypGetColItems(1, vtDimensionName, vtMembers)
End sub
```

HypGetConnectionInfo

Description

HypGetConnectionInfo() returns the connection information for the dynamic link query.

HypGetConnectionInfo assumes that a call has already been made to HypGetSourceGrid to initialize the dynamic link query, which contains the information about the active data source and the grid on the sheet.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Note:

After a call, the password is not actually returned, but for security reasons, is returned as empty.

Syntax

HypGetConnectionInfo(vtServerName, vtUserName, vtPassword, vtApplicationName, vtDatabaseName, vtFriendlyName, vtURL, vtProviderType)

ByRef vtServerName As Variant

ByRef vtUserName As Variant

ByRef vtPassword As Variant

ByRef vtApplicationName As Variant

ByRef vtDatabaseName As Variant ByRef vtFriendlyName As Variant ByRef vtURL As Variant ByRef vtProviderType As Variant

Parameters

vtServerName: Output. Contains the server name for the dynamic link query. vtUserName: Output. Contains the user name for the dynamic link query. vtPassword: The user password. vtApplicationName: Output. Contains the application name for the dynamic link query. vtDatabaseName: Output. Contains the database name for the dynamic link query. vtFriendlyName: Output. Contains the friendly connection name for the dynamic link query. vtURL: Output. Contains the URL for the dynamic link query.

vtProviderType: Output. Contains provider type for the dynamic link query.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

Declare Function HypGetConnectionInfo Lib "HsAddin.dll" (ByRef vtServerName As Variant, ByRef vtUserName As Variant, ByRef vtPassword As Variant, ByRef vtApplicationName As Variant, ByRef vtDatabaseName As Variant, ByRef vtFriendlyName As Variant, ByRef vtURL As Variant, ByRef vtProviderType As Variant) As Long

```
Sub Macro()
   Dim vtGrid as Variant
   Dim server As Variant
   Dim user As Variant
   Dim app As Variant
   Dim pass As Variant
   Dim db As Variant
   Dim provider As Variant
   Dim conn As Variant
   Dim url As Variant
   Sts = HypConnect("Sheet1", "system", "MyDemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypGetConnectionInfo(server, user, pass, app, db, conn, url,
provider)
End sub
```

HypGetCurrentAPSURL

Description

HypGetCurrentAPSURL gets the current Provider Services URL to be used (also shown in the Options dialog box). It returns:

- The global Provider Services URL, if the override flag is not set
- Current user's Provider Services URL, if the override flag is set

Syntax

```
HypGetCurrentAPSURL (vtAPSURL As Variant)
ByRef vtAPSURL As Variant
```

Parameters

vtAPSURL: the current Provider Services URL

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypGetCurrentAPSURL Lib "HsAddin.dll" (ByRef vtAPSURL As
Variant) As Long
Sub SubHypGetOverrideTest()
        Dim 1Ret As Long
        Dim bOverride As Boolean
        lRet = HypGetOverrideFlag(bOverride)
End Sub
```

HypGetDataPoint

Description

HypGetDataPoint() retrieves member information for a single data cell. For example, to find out the members that consist of the data intersection at cell B6, HypGetDataPoint may return the members January, California, Actual, Root Beer, Profit.

Syntax

HypGetDataPoint (vtSheetName, vtCell)

By Val vtSheetName As Variant

By Val vtCell As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtCell: Cell (range) that describes the reference cell for which to retrieve the member combination information.

Return Value

Returns an array of member names.

Example

```
Declare Function HypGetDataPoint Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal cell As Variant) As Variant
Sub DataPointsSub()
Dim vt As Variant
Dim cbItems As Variant
Dim i As Integer
Dim pMember As String
vt = HypDataPoint(Empty, "B3")
If IsArray(vt) Then
    cbItems = UBound(vt) - LBound(vt) + 1
    MsgBox ("Number of elements = " + Str(cbItems))
    For i = LBound(vt) To UBound(vt)
       MsgBox ("Member = " + vt(i))
    Next
    X = HypFreeDataPoint(vt)
MsgBox ("Return Value = " + Str(vt))
End If
End Sub
```

HypGetGlobalOption

Description

HypGetGlobalOption() returns information about individual Smart View workspace options.

Note:

For option descriptions, see Chapter 16, "User Preferences."

Syntax

HypGetGlobalOption(vtItem)

ByVal vtItem As Long

Parameters

vtItem: Number that indicates which option is to be retrieved.

Table 15 lists the numbers of options and their return data types.

Table 15

| vtItem | Option | Return Data Type |
|--------|---|------------------|
| 1 | Enable Excel formatting | Boolean |
| 2 | Enable double-click for Ad Hoc operations | Boolean |
| 3 | Enable Undo | Boolean |
| 4 | Not used | Boolean |
| 5 | Specify message level setting: | Number |
| | O Information messages | |
| | 1 Warning messages | |
| | 2 Error messages | |
| | 3 No messages | |
| 6 | Use thousands separator | Boolean |
| 7 | Enable route messages to log file | Boolean |
| 8 | Clear log file on next launch | Boolean |
| 9 | Enable Navigate Without Data | Boolean |
| | | |

Return Value

Returns a number or Boolean value indicating the state of the requested option. Returns an error code if parameter item is out of range.

Example

The following example sets the option for specifying a message level and checks whether the value set is valid.

```
Declare Function HypGetGlobalOption Lib "HsAddin.dll" (ByVal vtItem As
Long) As Variant
Sub GetGlobal()
   sts = HypGetGlobalOption(5)
```

```
If sts = -15 then
     Msgbox ("Invalid Parameter")
     Msgbox ("Message level is set to" & sts)
End Sub
```

HypGetLinkMacro

Description

HypGetLinkMacro() returns the macro name currently set to be run to perform the dynamic link query.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypGetLinkMacro (vtMacroName)

ByRef vtMacroName As Variant

Parameters

vtMacroName: Output. Returns the currently set macro name.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

```
Declare Function HypGetLinkMacro Lib "HsAddin.dll" (ByRef vtMacroName As
Variant) As Long
Sub Auto_Open()
   Dim Macroname as Variant
   Sts = HypUseLinkMacro(True)
   Sts = HypSetLinkMacro("Sheet1.Macro8")
   Sts = HypGetLinkMacro(Macroname)
   If (StrComp(MacroName, "Sheet1.Macro8")) Then
     MsgBox ("Error Occurred")
   End If
End Sub
```

HypGetOverrideFlag

Description

HypGetOverrideFlag gets the override flag value from Hyperion Options. The value of this flag indicates which Provider Services URL will be used. Possible values returned in vtOverride:

- True the urrent user's Provider Services URL will be used to try to connect to Provider Services
- False the global Provider Services URL will be used to try to connect to Provider Services

Syntax

```
HypGetOverrideFlag(vtOverride)
ByRef vtOverride As Boolean
```

Parameters

vtOverride

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypGetOverrideFlag Lib "HsAddin.dll" (ByRef vtOverride As
Boolean) As Long
Sub SubHypGetOverrideTest()
        Dim 1Ret As Long
        Dim bOverride As Boolean
        lRet = HypGetOverrideFlag(bOverride)
End Sub
```

HypGetParent

Description

HypGetParent() returns the parent name for the specified member.

Syntax

```
HypGetParent(vtSheetName, vtMemberName, vtParentName)
```

ByVal vtSheetName As Variant

ByVal vtMemberName As Variant

ByRef vtParentName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtMemberName: Specify a member name. Required Field.

vtParentName: Output. Contains the parent name on successful execution of the macro.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypGetParent Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtMemberName As Variant, ByRef vtParentName As Variant) As
Long
Sub Sample_HypGetParent
   Dim vtParent as Variant
   X = HypGetParent ("Sheet1", "East", vtParent)
End sub
```

HypGetPOVCount

Description

HypGetPOVCount() returns the number dimensions in the POV from the dynamic link query.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypGetPOVCount()

Return Value

Returns the number of column dimensions if successful; otherwise, returns the negative error code.

Example

```
Declare Function HypGetPOVCount Lib "HsAddin.dll" () As Long

Sub Macro()
   Dim vtGrid as Variant
   Sts = HypConnect("Sheet1", "system", "password", "MyDemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypGetPOVCount ()
```

HypGetPOVItems

Description

HypGetPOVItems() returns the dimensions in the POV and the currently selected member for each dimension.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypGetPOVItems(vtDimensionNames, vtPOVNames) ByRef vtDimensionNames As Variant

ByRef vtPOVNames As Variant

Parameters

vtDimensionNames: The dimension names in the POV

vtPOVNames: The currently selected member for each dimension in the POV.

Return Value

Returns 0 if successful; otherwise, returns the negative error code.

```
Declare Function HypGetPOVItems Lib "HsAddin.dll" (ByRef vtDimensionNames As Variant, ByRef vtPOVNames As Variant) As Long

Sub Macro()

Dim vtGrid as Variant

Dim vtDimNames As Variant
```

```
Dim vtPOVNames As Variant
   Sts = HypConnect("Sheet1", "system", "password", "MyDemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypGetPOVItems (vtDimNames, vtPOVNames)
End sub
```

HypGetRowCount

Description

HypGetRowCount() returns the number of row dimensions.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypGetRowCount()

Return Value

Returns number of row dimensions if successful; otherwise, returns the negative error code.

Example

```
Declare Function HypGetRowCount Lib "HsAddin.dll" () As Long
Sub Macro()
  Dim vtGrid as Variant
   Sts = HypConnect("Sheet1", "system", "password", "MyDemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypGetRowCount ()
End sub
```

HypGetRowItems

Description

HypGetRowItems() returns the members present for the nth row dimension in the dynamic link query.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypGetRowItems(vtRowID, vtDimensionName, vtMemberNames)

ByVal vtRowID As Variant

ByRef vtDimensionName As Variant

ByRef vtMemberNames As Variant

Parameters

vtRowID: The row number n.

vtDimensionName: Returns the nth row dimension name.

vtMemberNames: Returns the members for the nth row dimensions.

Return Value

Returns 0 if successful; otherwise, returns the negative error code.

Example

```
Declare Function HypGetRowItems Lib "HsAddin.dll" (ByVal rowID As Variant,
ByRef vtDimensionName As Variant, ByRef vtMemberNames As Variant) As Long
Sub Macro()
   Dim vtGrid as Variant
  Dim vtDimName as Variant
  Dim vtMembers as Variant
   Sts = HypConnect("Sheet1", "system", "password", "AnamikaDemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypGetRowItems(1, vtDimName, vtMembers)
End sub
```

HypGetSheetOption

Description

HypGetSheetOption() returns information about individual spreadsheet options.

Syntax

HypGetSheetOption(vtSheetName, vtItem)

ByVal vtSheetName As Variant ByVal vtItem As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtItem: Number indicating which option is to be retrieved. See Table 16 for a list of values.

Table 16 indicates which options are returned for the vtItem parameter.

Table 16

| vtItem | Option | Data Type and Values |
|--------|---|----------------------|
| 1 | Set zoom in level: | Number |
| | 0 Next level | |
| | 1 All levels | |
| | 2 Bottom level | |
| 2 | Enable Include Selection setting | Boolean |
| 3 | Enable Within Selection Group setting | Boolean |
| 4 | Enable Remove Unselected Groups setting | Boolean |
| 5 | Specify Indent setting: | Number |
| | 0 No indentation | |
| | 1 Indent sub items | |
| | 2 Indent totals | |
| 6 | Enable suppress missing setting | Boolean |
| 7 | Enable suppress zeros setting | Boolean |
| 8 | Enable suppress underscores setting | Boolean |
| 9 | Enable No Access setting | Boolean |
| 10 | Enable Repeated Members setting | Boolean |
| 11 | Enable invalid setting | Boolean |
| 12 | Ancestor Position: | Number |
| | 0 Тор | |
| | 1 Bottom | |
| 13 | Specify Missing text label | Text |

| vtItem | Option | Data Type and Values |
|--------|---|----------------------|
| 14 | Specify No Access label | Text |
| 15 | Cell status: 0 Data 1 Calculation status 2 Process Management | Number |
| 16 | Member Name Display options: 0 Name only 1 Name and Description 2 Description only | Number |

Return Value

Returns the value of the current setting as a string, number, or Boolean. Returns an error code if parameter item is out of range.

Example

```
Declare Function HypGetSheetOption Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtItem As Variant) As Variant
Sub GetSheet()
```

```
sts = HypGetSheetOption("Sheet", 5)
If sts = -15 then
  Msgbox ("Invalid Parameter")
   Msgbox ("Indentation is set to" & sts)
End Sub
```

HypGetSourceGrid

Description

HypGetSourceGrid() creates a query from the source grid for the dynamic link query.

This function applies to both static and dynamic link views.

Note:

A cell in the grid must be selected before this making this call.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypGetSourceGrid(vtSheetName, vtGrid)

ByVal vtSheetName As Variant

ByRef vtGrid As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtGrid: The grid XML is returned on successful execution.

Return Value

Returns 0 if successful or the appropriate error code otherwise.

Example

```
Declare Function HypGetSourceGrid Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByRef vtGrid As Variant) As Long
Sub Macro()
   Dim vtGrid as Variant
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
End sub
```

HypGetSubstitutionVariable

Description

HypGetSubstitutionVariable() retrieves substitution variables and their current value from Analytic Server.

Syntax

HypGetSubstitutionVariable (vtSheetName, vtApplicationName, vtDatabaseName, vtVariableNameList, vtVariableValueList)

ByVal vtSheetName As Variant

ByVal vtApplicationName As Variant

ByVal vtDatabaseName As Variant

ByVal vtVariableName As Variant

ByRef vtVariableNameList As Variant

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtApplicationName: The application name to return variables scoped for the specified application. If vtApplicationName is Null or Empty all the applications are considered.

vtDatabaseName: The database name to return variables scoped for the specified database. If vtDatabaseName is Null or Empty all the databases are considered.

vtVariableName: The variable name to be retrieved. If vtVariableName is Null or Empty the entire list of variables is returned.

vtVariableNameList: Output Result Vector that contains the list of the variable names. Its contents are unknown if the macro fails.

vtVariableValueList: Output Result Vector that contains the list of the variable values corresponding to each variable returned. Its contents are unknown if the macro fails.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

Declare Function HypGetSubstitutionVariable Lib "HsAddin.dll" (ByVal vtSheetName As Variant, ByVal vtApplicationName As Variant, ByVal vtDatabaseName As Variant, ByVal vtVariableName As Variant, ByRef vtVariableNames As Variant, ByRef vtVariableValues As Variant) As Long

```
Sub Sample_HypGetSubstitutionVariable
  Dim vtVarNameList as Variant
  Dim vtVarValueList as Variant
  Dim vtVarVal as Variant
  Dim vtVar as Variant
  X = HypGetSubstitutionVariable ("Sheet1", "Sample", "Basic",
   Empty, vtVarNameList, vtVarValueList)
      If IsArray (vtVarNameList) Then
         For i = LBound (vtVarNameList) To UBound (vtVarNameList)
         vtVar = vtVarNameList (i)
        Next.
      End If
      If IsArray (vtVarValueList) Then
         For i = LBound (vtVarValueList) To UBound (vtVarValueList)
         vtVarVal = vtVarValueList (i)
        Next
   End If
```

HypIsAttribute

Description

HypIsAttribute() checks to see if the specified member has a specific attribute.

Syntax

HypIsAttribute(vtSheetName, vtDimensionName, vtMemberName, vtAttributeName)

ByVal vtSheetName As Variant

ByVal vtDimensionName As Variant

ByVal vtMemberName As Variant

ByVal vtAttributeName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls]Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtDimensionName: The name of the dimension where the member belongs.

vtMemberName: The name of the member for which we must test the condition.

vtAttributeName: Input string that is compared against the attributes of the member.

Return Value

Returns a variant in which -1 is true, 0 is false; otherwise, returns the appropriate error code.

```
Declare Function HypIsAttribute Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtDimensionName As Variant, ByVal vtMemberName As Variant,
ByVal vtAttribute As Variant) As Variant

Sub CheckAttribute()
vtret = HypIsAttribute("Sheet1", "Market", "Connecticut", "MyAttribute")
    If vtret = -1 Then
        MsgBox ("Found MyAttribute")
    ElseIf vtret = 0 Then
        MsgBox ("MyAttribute not available for Connecticut")
    Else
        MsgBox ("Error value returned is" & vtret)
    End If
End Sub
```

HypIsConnectedToAPS

Description

HypIsConnectedToAPS checks whether Smart View is connected to Provider Services.

Syntax

HypIsConnectedToAPS()

Return Value

Return: true if Smart View is connected to Provider Services, otherwise false.

Example

```
Declare Function HypIsConnectedToAPS Lib "HsAddin.dll" () As Variant
Sub SubIsAPSConnectedTest()
        Dim vtRet As Variant
        vtRet = HypIsConnectedToAPS()
End Sub
```

HypIsDescendant

Description

HypIsDescendant() checks if the specified member is the descendant of another specified member.

Syntax

HypIsDescendant(vtSheetName, vtMemberName, vtAncestorName)

ByVal vtSheetName As Variant

ByVal vtMemberName As Variant

ByVal vtAncestorName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtMemberName: A member name. Required.

vtAncestorName: The member name of the ancestor. Required.

Return Value

Returns a variant in which -1 is true, 0 is false; otherwise, returns the appropriate error code.

Example

```
Declare Function HypIsDescendant Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtMemberName As Variant, ByVal vtDescendantName As Variant)
As Boolean
Sub Sample HypIsDescendant
   Dim b as Boolean
   b = HypIsDescendant ("Sheet1", "Year", "Jan")
```

HypIsExpense

Description

HypIsExpense() verifies that the member specified has an Expense tag.

Syntax

HypIsExpense(vtSheetName, vtDimensionName, vtMemberName)

ByVal vtSheetName As Variant

ByVal vtDimensionName As Variant

ByVal vtMemberName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtDimensionName: The name of the dimension where the member belongs. If vtDimensionName is Null or Empty, the active dimension is used.

vtMemberName: The name of the specified member.

Return Value

Returns a variant in which -1 is true, 0 is false; otherwise, returns the appropriate error code.

Example

Declare Function HypIsExpense Lib "HsAddin.dll" (ByVal vtSheetName As Variant, ByVal vtDimensionName As Variant, ByVal vtMemberName As Variant) As Variant

```
Sub CheckExpense()
vtret = HypIsExpense("Sheet1", "Measures", "Opening Inventory")
    If vtret = -1 Then
    MsgBox ("Opening Inventory has expense flag set")
    ElseIf vtret = 0 Then
    MsgBox ("Expense flag has not been set")
    MsgBox ("Error value returned is" & vtret)
    End If
End Sub
```

HypIsParent

HypIsParent() checks if the specified member is the parent of another specified member.

Syntax

HypIsParent(vtSheetName, vtMemberName, vtParentName)

ByVal vtSheetName As Variant

ByVal vtMemberName As Variant

ByVal vtParentName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtMemberName: A member name. Required.

vtParentName: The member name of the parent. Required.

Return Value

Returns a variant in which -1 is true, 0 is false; otherwise, returns the appropriate error code.

```
Declare Function HypIsParent Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtMemberName As Variant, ByVal ParentName As Variant) As
Boolean
Sub Sample_HypIsParent
   Dim b as Boolean
   b = HypIsParent ("sheet1", "East", "Market")
End Sub
```

HypIsUDA

Description

HypIsUDA() checks to verify if the member specified has a specific UDA.

Syntax

HypIsUDA (vtSheetName, vtDimensionName, vtMemberName, vtUDAString)

ByVal vtSheetName As Variant

ByVal vtDimensionName As Variant

ByVal vtMemberName As Variant

ByVal vtUDAString As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtDimensionName: The name of the dimension where the member belongs.

vtMemberName: The name of the member for which we must test the condition.

vtUDAString: Input string that is compared against the attributes of the member.

Return Value

Returns a variant in which -1 is true, 0 is false; otherwise, returns the appropriate error code.

```
Declare Function HypIsUDA Lib "HsAddin.dll" (ByVal vtSheetName As Variant,
ByVal vtDimensionName As Variant, ByVal vtMemberName As Variant, ByVal
vtUDAString As Variant) As Variant
```

```
Sub CheckUDA()
vtret = HypIsUDA("Sheet1", "Market", "Connecticut", "MyUDA")
    If vtret = -1 Then
     MsgBox ("Found MyUDA")
    ElseIf vtret = 0 Then
     MsgBox ("Did not find MyUDA")
     MsgBox ("Error value returned is" & vtret)
    End If
End Sub
```

HypKeepOnly

Description

HypKeepOnly() retains only the selected member(s) in the sheet and removes unselected members.

The selection must be limited to member cells only, not data cells.

Syntax

HypKeepOnly(vtSheetName, vtSelection)

ByVal vtSheetName As Variant

ByVal vtSelection As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtSelection: Range object which refers to the member(s) that will be kept. If selection is Null or Empty, the active cell is used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

```
Declare Function HypKeepOnly Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtSelection As Variant) As Long
Sub KOnly()
' Keep Only on one member name
  X=HypKeepOnly("[Book2.xls]Sheet1", RANGE("D2"))
   If X = 0 Then
     MsgBox("Keep Only successful.")
   Else
     MsgBox("Keep Only failed." + X)
' Keep Only on two member names
  X=HypKeepOnly("[Book2.xls]Sheet1", RANGE("D2:A5"))
   If X = 0 Then
     MsgBox("Keep Only successful.")
      MsgBox("Keep Only failed." + X)
   End If
End Sub
```

HypListCalcScripts

Description

HypListCalcScripts() lists all calculation scripts present on Analytic Server.

Syntax

```
HypListCalcScripts (vtSheetName, vtScriptArray)
ByVal vtSheetName As Variant
ByRef vtScriptArray As Variant
```

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls]Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtScriptArray: The business rule scripts are returned in this array.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypListCalcScripts Lib "HsAddin.dll" (ByVal sheetName As
Variant,ByRef scriptArray) As Long

Dim sts As Long
Dim paramList As Variant
sts=HypListCalcScripts ("sheet1",paramList)
If IsArray(paramList) Then
   cbItems = UBound(paramList) - LBound(paramList) + 1
        MsgBox ("Number of elements = " + Str(cbItems))
For i = LBound(paramList) To UBound(paramList)
        MsgBox ("Member = " + paramList(i))
Next
Else
        MsgBox ("Return Value = " + sts))
End If
```

HypLoginSetPassword

Description

HypLoginSetPassword() sets the password upon login, then logs the user out.

Syntax

HypLoginSetPassword (vtSheetName, vtNewPassword, vtOldPassword, vtServerName, vtUserName)

ByVal vtSheetName As Variant

ByVal vtNewPassword As Variant

ByVal vtOldPassword As Variant

ByVal vtServerName As Variant

ByVal vtUserName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtNewPassword: Text name of the new password you want to set for the user name.

vtOldPassword: Text name of the old password that you want to replace for the user name.

vtServerName: Text name of the server you want to change the password for.

vtUserName: Text name of the valid user name on the server.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

Declare Function HypLoginSetPassword Lib "HsAddin.dll" (ByVal vtSheetName as Variant, ByVal vtNewPassword As Variant, ByVal vtOldPassword As Variant, ByVal vtServerName As Variant, ByVal vtUserName As Variant) As Long

```
Sub SetPassword()
Dim X As Long
'This sets the login password to password2 from password1 for the
'user User1 on server Local.X
=HypLoginSetPassword ("[Budget.xls]Sheet1", "password2", "password1",
"Local", "User1")
If X=0 then
  MsgBox("Set Password Successful.")
  MsgBox("Set Password Unsuccessful.")
End If
End Sub
```

HypMigrateConnectionToDataSourceMgr

Description

HypMigrateConnectionToDataSourceMgr helps you migrate connections to the Data Source Manager.

Syntax

```
\label{thm:main} Hyp Migrate Connection To Data Source Mgr (vt Conn Friendly Name) \\ vt Conn Friendly Name
```

Parameters

vtConnFriendlyName: The connection name

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

HypOtlGetMemberInfo

Description

HypOtlGetMemberInfo() returns the following information related to a member selection: member comment, formula, UDA, attribute, etcetera.

Syntax

HypOtlGetMemberInfo (vtSheetName, vtDimensionName, vtMemberName, vtPredicate, vtMemberArray)

ByVal vtSheetName As Variant

ByVal vtDimensionName As Variant

ByVal vtMemberName As Variant

ByVal vtPredicate As Variant

ByRef vtMemberArray As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtDimensionName: The name of the dimension. Can be Null; if Null, search for the predicate in the whole outline. Dimension to limit the scope of the query.

vtMemberName: Member name for which information is being queried on.

vtPredicate: Member selection criteria:

```
1 HYP_COMMENT
2 HYP_FORMULA
3 HYP_UDA
4 HYP_ATTRIBUTE
```

vtMemberArray: Output that contains the result of the query. Its contents are unknown if the macro fails.

Return Value

Returns 0 if successful; otherwise returns the appropriate error code.

Example

Declare Function HypOtlGetMemberInfo Lib "HsAddin.dll" (ByVal vtSheetName As Variant, ByVal vtMemberName As Variant, ByVal vtPredicate As Variant, ByRef vtMemberArray As Variant) As Long

```
Sub HypOtlGetMemberInfo()
    vtRet = HypOtlGetMemberInfo (vtSheetName, "Year", "Jan",
    HYP_COMMENT, vt)
If IsArray(vt) Then cbItems = UBound(vt) + 1
   MsgBox ("Number of elements = " + Str(cbItems))
For i = 0 To UBound(vt)
   MsgBox ("Member = " + vt(i))
    Next
Else
MsgBox ("Return Value = " + vtRet)
End If
End Sub
```

HypPivot

Description

HypPivot() transposes spreadsheet rows and columns, based on the selected dimension.

Syntax

```
HypPivot(vtSheetName, vtStart, vtEnd)
ByVal vtSheetName As Variant
ByVal vtStart As Variant
ByVal vtEnd As Variant
```

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls]Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtStart: Range object which refers to the single cell starting point of the pivot.

vtEnd: Range object which refers to the single cell ending point of the pivot

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypPivot Lib "HsAddin.dll" (ByVal vtSheetName As Variant,
ByVal vtStart As Variant, ByVal vtEnd As Variant) As Long

Sub DoPivot()
X=HypPivot("[Book2.xls]Sheet1", RANGE("B2"), RANGE("D1"))
   If X = 0 Then
        MsgBox("Pivot successful.")
   Else
        MsgBox("Pivot failed.")
   End If
End Sub
```

HypPivotToGrid

Description

HypPivotToGrid() moves the selected dimension and members from the POV to the spreadsheet grid.

Syntax

HypPivotToGrid (vtSheetName, vtDimensionName, vtSelection)

ByVal vtSheetName as Variant

ByVal vtDimensionName as Variant

ByVal vtSelection as Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtDimensionName: Currently selected dimension from the toolbar.

vtSelection: Range object which refers to the single cell starting point of the pivot. Orientation is calculated based on the selection.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypPivotToGrid Lib "HsAddin.dll" (By Val vtSheetName As
Variant, ByVal vtDimensionName as Variant, ByVal vtSelection as Variant) As
Long
Sub DoPivotGrid()
X=HypPivotToGrid("[Book2.xls]Sheet1", "Product", RANGE("E6"))
If X = 0 Then
   MsgBox("Pivot to grid successful.")
   MsgBox("Pivot to grid failed.")
End If
End Sub
```

HypPivotToPOV

Description

HypPivotToPOV() pivots from the spreadsheet grid to the POV.

Syntax

HypPivotToPOV (vtSheetName, vtSelection)

ByVal vtSheetName as Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtSelection: Range object which refers to the single cell starting point of the pivot. Orientation is calculated based on the selection.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypPivotToPOV Lib "HsAddin.dll" (By Val vtSheetName As
Variant, ByVal vtSelection as Variant) As Long
Sub DoPivotPOV()
X=HypPivotToPOV("[Book2.xls]Sheet1", RANGE("E6"))
If X = 0 Then
   MsgBox("Pivot to POV successful.")
   MsgBox("Pivot to POV failed.")
End If
End Sub
```

HypQueryMembers

Description

HypQueryMembers() executes the member selection query.

Syntax

HypQueryMembers (vtSheetName, vtMemberName, vtPredicate, vtOption, vtDimensionName, vtInput1, vtInput2, vtMemberArray)

ByVal vtSheetName As Variant

ByVal vtMemberName As Variant

ByVal vtPredicate As Variant

ByVal vtOption As Variant

ByVal vtDimensionName As Variant

ByVal vtInput1 As Variant

ByVal vtInput2 As Variant

ByRef vtMemberArray As Variant

Parameters

vtSheetName: (string) Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtMemberName: (string) The member name on which to perform the query.

vtPredicate: (integer) Member selection criteria:

1 HYP CHILDREN

2 HYP_DESCENDANTS

3 HYP_BOTTOMLEVEL

4 HYP SIBLINGS

5 HYP_SAMELEVEL

6 HYP_SAMEGENERATION

7 HYP_PARENT

8 HYP_DIMENSION

9 HYP_NAMEDGENERATION

10 HYP_NAMEDLEVEL

11 HYP_SEARCH

12 HYP_WILDSEARCH

13 HYP_USERATTRIBUTE

14 HYP_ANCESTORS

15 HYP_DTSMEMBER

16 HYP_DIMUSERATTRIBUTES

vtOption: (integer) Options are dependent on the predicate:

For the predicate values, HYP_SEARCH and HYP_WILDSEARCH, specify query options: **HYP_MEMBERSONLY**

HYP_ALIASESONLY

HYP_MEMBERSANDALIASES

vtDimensionName: (string) Dimension to limit the scope of the query. It is used with the following query options and ignored otherwise: HYP_NAMEDGENERATION, HYP_NAMEDLEVEL, HYP_USERATTRIBUTE HYP_SEARCH (set to Null to search through all dimensions), HYP WILDSEARCH (set to Null to search through all dimensions).

vtInput1: (string) Input string that is determined by the option. It is used with the following query options and ignored otherwise:

- HYP NAMEDGENERATION (The name of the generation)
- HYP_NAMEDLEVEL (The name of the level)
- HYP_SEARCH (The string to search for. The string is defined as an exact)
- HYP WILDSEARCH (The string to search for. The string is defined as an exact search string with an optional '*' at the end to mean any set of characters)
- HYP USERATTRIBUTE (The user-defined attribute)

vtInput2: (string) Input string that is determined by the option. It is used with the following query options and ignored otherwise:

- HYP USERATTRIBUTE (The user-defined attribute)
- HYP_SEARCH, HYP_WILDSEARCH (If the options are set to search in the alias tables, this string specifies which alias table to search. If the string is Null, all alias tables will be searched).

vtMemberArray: Output that contains the result of the query. If unsuccessful, its contents are unknown.

Return Value

Returns a zero if successful; otherwise, returns the appropriate error code.

Example

Declare Function HypQueryMembers Lib "HsAddin.dll" (ByVal vtSheetName As Variant, ByVal vtMemberName As Variant, ByVal vtPredicate As Variant, ByVal vtOption As Variant, ByVal vtDimensionName As Variant, ByVal vtInput1 As Variant, ByVal vtInput2 As Variant, ByRef vtMemberArray As Variant) As Long

```
Sub QueryMembersEmptyValues()
 vtRet = HypQueryMembers (Empty, Null, HYP_WILDSEARCH,
   HYP_MEMBERSONLY, "Year", "J*", "", vt)
If IsArray(vt) Then
  cbItems = UBound(vt) + 1
      MsgBox ("Number of elements = " + Str(cbItems))
  For i = 0 To UBound(vt)
      MsgBox ("Member = " + vt(i))
  Next.
Else
  MsgBox ("Return Value = " + Str(vt))
End If
End Sub
```

HypRedo

Description

HypRedo() restores the database view as it was before an Undo was performed.

Syntax

HypRedo (vtSheetName)

ByVal vtSheetName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

Return Value

Returns 0 if successful. A negative number indicates a local failure. A return value greater than zero indicates a failure on the server.

Example

```
Declare Function HypRedo Lib "HsAddin" (ByVal vtSheetName As Variant) As
Long
Sub Redo()
  X=HypRedo("[Book2.xls]Sheet1")
```

HypRemoveConnection

Description

HypRemoveConnection() removes the specified connection from the list of all available Smart View connections in the Data Source Manager.

Syntax

HypRemoveConnection(vtFriendlyName)

ByVal vtFriendlyName As Variant

Parameters

vtFriendlyName: The friendly connection name for the data source provider

Return Value

Returns 0 if successful, otherwise, returns the appropriate error code.

Example

```
Declare Function HypRemoveConnection Lib "HsAddin" (ByVal vtFriendlyName
As Variant) As Long
Sub RConn()
   X=HypRemoveConnection("My Connection")
End Sub
```

HypRemoveOnly

Description

HypRemoveOnly() removes only the selected member(s) in the sheet and retains unselected members in the selected dimension.

Selection should include only member cells, not data cells.

Syntax

HypRemoveOnly(vtSheetName, vtSelection) ByVal vtSheetName As Variant ByVal vtSelection As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtSelection: Range object which refers to the member(s) that will be removed. If selection is Null or Empty, the active cell is used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypRemoveOnly Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtSelection As Variant) As Long
Sub ROnly()
' Remove Only on one member name
X=HypRemoveOnly("[Book2.xls]Sheet1", RANGE("D2"))
If X = 0 Then
   MsgBox("Remove Only successful.")
Else
   MsgBox("Remove Only failed." + X)
End If
```

```
' Remove Only on two member names
X=HypRemoveOnly("[Book2.xls]Sheet1", RANGE("D2, A5"))
If X = 0 Then
  MsgBox("Remove Only successful.")
  MsgBox("Remove Only failed." + X)
End If
End Sub
```

HypResetFriendlyName

Description

HypResetFriendlyName resets the friendly name to the new friendly name if the new name does not exist. To modify friendly name of a connection in the Data Source Manager, Smart Viewmust be connected to Provider Services.

Syntax

HypResetFriendlyName (vtOldFriendlyName, vtNewFriendlyName)

By Val vtOldFriendlyName as Variant

By Val vtNewFriendlyName as Variant

Parameters

vtOldFriendlyName: The old friendly connection name.

vtNewFriendlyName: The new friendly connection name.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypResetFriendlyName Lib "HsAddin.dll" (ByVal
vtOldFriendlyName As Variant, ByVal vtNewFriendlyName As Variant) As Long
Sub SubHypResetFriendlyNameTest()
        Dim 1Ret As Long
        1Ret = HypResetFriendlyName("server2 Sample Basic", "My Sample
Basic")
End Sub
```

HypRetrieve

Description

HypRetrieve() retrieves data from the database.

Syntax

```
HypRetrieve(vtSheetName)
ByVal vtSheetName As Variant
```

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypRetrieve Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtRange As Variant, ByVal vtLock As Variant) As Long
Sub RetData()
X=HypRetrieve("[Book2.xls]Sheet1")
If X = 0 Then
  MsgBox("Retrieve successful.")
Else
   MsgBox("Retrieve failed.")
End If
End Sub
```

HypRetrieveRange

Description

HypRetrieveRange() gives users the ability to refresh a selected or named range of cells in a grid or worksheet. If the range provided to this function contains more rows or columns than the actual grid has, the additional rows and columns are treated as comments and are thus part of the grid.

Range retrieval clears the Undo buffer, therefore the Undo operation cannot be used afterward.

Syntax

HypRetrieveRange(vtSheetName,vtRange,vtConnName)

ByVal vtSheetName As Variant

ByVal vtRange As Variant

ByVal vtConnectionName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Single continuous range to be refreshed. If vtRange is Null, the entire worksheet is refreshed, and GetUsedRange is used on the worksheet specified to get the range to be refreshed.

vtConnectionName: Friendly name of the connection to be used to refresh the range. If vtConn is Null, the active connection associated with the worksheet is used to refresh the range on that worksheet. If no connection is associated, an error is returned.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypRetrieveRange Lib "HsAddin" (ByVal vtSheetName As
Variant, ByVal vtRange As Variant, ByVal vtConnName As Variant) As Long
Worksheets("Sheet2").Names.Add name:="MyRange", RefersTo:="=$E$11:$F$28"
Sub Sample_RetrieveRange
    Worksheets("Sheet2").Names.Add name:="MyRange", RefersT o:="=$E$11:
$
             F$28"
    sts = HypRetrieveRange("Sheet2", range("E11:F28"), "Samp1")
            'retrieve by regular range
    sts = HypRetrieveRange("Sheet2", range("MyRange"), "Samp1")
            'retrieve by named range
End sub
```

HypSetActiveConnection

Description

HypSetActiveConnection() is used to associate the current active worksheet with one of the active connections.

Note:

HypSetActiveConnection does not work with worksheets that contain Report Designer objects

Syntax

HypSetActiveConnection (vtConnectionName)

ByVal vtConnectionName as Variant

Parameters

vtConnectionName: Name of the active connection which is to be associated with the current active worksheet. It is not case-sensitive.

Return Value

Long. If successful, return value is 0; otherwise, the appropriate error code is returned.

Example

```
Declare Function HypSetActiveConnection Lib "HsAddin.dll" (ByVal
vtConnectionName As Variant) As Long
Sub Sample_SetActiveConnection
   sts = HypSetActiveConnection ("Demo_Basic")
End sub
```

HypSetAliasTable

Description

HypSetAliasTable() enables users to change alias tables in Essbase

Syntax

HypSetAliasTable (ByVal vtSheetName As Variant, ByVal vtAliasTableName As Variant)

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtAliasTableName: Text name of the alias table. vtAliasTableName is of the form "Default", "Long Names" and so forth.

Return Value

0 if successul, else negative value

Example

```
Public Declare Function HypSetAliasTable Lib "HsAddin.dll" (ByVal
vtSheetName As Variant, ByVal vtAliasTableName As Variant) As Long
Sub Sample SetActiveConnection
  sts = HypSetAliasTable("Sheet1", "Long Name")
End sub
```

HypSetBackgroundPOV

Description

HypSetBackgroundPOV() sets the POV for the connection object in the POV Manager.

Syntax

HypSetBackgroundPOV(vtFriendlyName, ParamArray MemberList())

ByVal vtFriendlyName As Variant

ParamArray MemberList() As Variant

Parameters

vtFriendlyName:Connection name for the data source provider.

MemberList: A list of strings which describe the member combination for which a data value will be retrieved. If MemberList is Null or Empty, the top level value is used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypSetBackgroundPOV Lib "HsAddin" (ByVal vtFriendlyName,
ParamArray MemberList() As Variant) As Long
Sub SetBGPOV()
   X=HypSetBackgroundPOV ("My Connection", "Year#Qtr1", "Market#East")
End Sub
```

HypSetCellsDirty

Description

HypSetCellsDirty() marks selected data range dirty for submit data.

Syntax

HypSetCellsDirty (vtSheetName, vtRange)

ByVal vtSheetName As Variant

ByVal vtRange As Variant

Parameters

vtSheetName: Text name of worksheet to perform the action. vtSheetName is of the form "[Book.xls]Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Variant data range to be marked as dirty.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypSetCellsDirty Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtRange As Variant) As Long
Sub SetDirtyCells()
   X=HypSetCellsDirty ("Sheet1", Range ("A3:B3")
End Sub
```

HypSetColItems

Description

HypSetColItems() sets the members for the nth column dimension for the dynamic link query. If the nth column does not exist, a new column is appended.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypSetColItems (vtColumnID, vtDimensionName, ppMemberList())

ByVal vtColumnID As Variant

ByVal vtDimensionName As Variant

ParamArray ppMemberList() As Variant

Parameters

vtColumnID: The column number n.

vtDimensionName: The dimension name.

ppMemberList: The list of member names.

Return Value

Long. Returns 0 if successful, otherwise, returns the negative error code.

Example

```
Declare Function HypSetColItems Lib "HsAddin.dll" (ByVal vtColID As
Variant, ByVal vtDimensionName As Variant, ParamArray MemberList() As
Variant) As Long
Sub Macro()
   Dim vtGrid as Variant
   Sts = HypConnect("Sheet1", "system", "password", "SalesDemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypSetColItems (1, "Market", "East", "West", "South",
    "Central", "Market")
End sub
```

HypSetConnAliasTable

Description

HypSetConnAliasTable() enables users to change alias tables for an Essbase connection.

Syntax

HypSetConnAliasTable (ByVal vtConnName As Variant, ByVal vtAliasTableName As Variant)

Parameters

vtConnName: Text name of the connection. vtConnName is of the form "SampleBasic". If vtConnName is Null or Empty, it will return an error . The basic requirement for this function is that it should have an active connection. For an active connection only the Alias table can be changed

vtAliasTableName: Text name of the Alias table. vtAliasTableName can be of the form "Default", "Long Names", "None" and so forth. This parameter cannot be Null or Empty. If no Alias has to be applied then you can use the parameter "None" for that purpose.

Return Value

Returns 0 if successul; otherwise, a negative value.

Example

```
Public Declare Function HypSetConnAliasTable Lib "HsAddin.dll" (ByVal
vtConnName As Variant, ByVal vtAliasTableName As Variant) As Long
Sub Sample_SetAliasTableForConnection
   sts = HypSetConnAliasTable("SampleBasic", "Long Name")
End sub
```

HypSetConnectionInfo

Description

HypSetConnectionInfo() is used to modify the connection information in the query.

The parameters passed for HypSetConnectionInfo() should be match the connection information stored with that connection name.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypSetConnectionInfo (vtServerName, vtUserName, vtPassword, vtApplicationName, vtDatabaseName, vtFriendlyName, vtURL, vtProviderType)

ByVal vtServerName As Variant

ByVal vtUserName As Variant

ByVal vtPassword As Variant

ByVal vtApplicationName As Variant

ByVal vtDatabaseName As Variant

ByVal vtFriendlyName As Variant

ByVal vtURL As Variant

ByVal vtProviderType As Variant

Parameters

vtServerName: The server name in the query.

```
vtUserName: The user name in the query.
vtPassword: The user password in the query.
vtApplicationName: The application name in the query.
vtDatabaseName: The database name in the query.
vtFriendlyName: The friendly connection name in the query.
```

vtURL: The provider URL in the query.

vtProviderType: The provider type in the query.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

Declare Function HypSetConnectionInfo Lib "HsAddin.dll" (ByVal vtServerName As Variant, ByVal vtUserName As Variant, ByVal vtPassword As Variant, ByVal vtApplicationName As Variant, ByVal vtDatabaseName As Variant, ByVal vtFriendlyName As Variant, ByVal vtURL As Variant, ByVal vtProviderType As Variant) As Long

```
Sub Macro()
  Dim vtGrid as Variant
  Sts = HypConnect("Sheet1", "system", "password", "DemoBasic")
  Sts = HypRetrieve("Sheet1")
  Range ("B2").Select
  Sts = HypGetSourceGrid ("Sheet1", vtGrid)
  Sts = HypSetConnectionInfo("localhost", "system",
    "password", "Sample", "Basic", "SampleBasic",
    "http://localhost:13080/aps/SmartView", provider)
End sub
```

HypSetCurrentUserAPSURL

Description

HypSetCurrentUserAPSURL sets the active user's Provider Services URL in the config file and Options dialog box. Sets the Override flag to true and specifies the Provider Services URL to be used.

Syntax

```
HypSetCurrentUserAPSURL (vtDefaultURL As Variant)
ByVal vtAPSURL As Variant)
```

Parameters

vtDefaultURL: the current the current Provider Services URL

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

HypSetGlobalOption

Description

HypSetGlobalOption() sets individual workspace options. For option descriptions, see Chapter 16, "User Preferences."

Note:

You can set only one option at a time.

Syntax

HypSetGlobalOption(vtItem, vtGlobalOption) ByVal vtItem As Long ByVal vtGlobalOption As Variant

Parameters

vtItem: Number indicating which option is to be retrieved. See Table 17 for values.

vtGlobalOption: A Boolean or Number value denoting the option being set for vtItem. If vtGlobalOption is Null or Empty, no action is performed.

Table 17 indicates which options are returned for the vtItem parameter.

Table 17 Values for HypSetGlobalOption item Parameter

| vtItem | Option | Return Data Type |
|--------|---|------------------|
| 1 | Enable Excel formatting | Boolean |
| 2 | Enable double-click for Ad Hoc operations | Boolean |
| 3 | Enable Undo | Boolean |
| 4 | Not used | Boolean |
| 5 | Specify message level setting: | Number |
| | 0 Information messages | |
| | 1 Warning messages | |
| | 2 Error messages | |
| | 3 No messages | |
| 6 | Use thousands separator | Boolean |
| 7 | Enable route messages to log file | Boolean |
| 8 | Clear log file on next launch | Boolean |
| 9 | Enable Navigate Without Data | Boolean |

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

The following example sets the option to display error messages only.

```
Declare Function HypSetGlobalOption Lib "HsAddin.dll" (ByVal vtItem As
Long, ByVal vtGlobalOption As Variant) As Long
Sub SetGlobal()
  X=HypSetGlobalOption(5, 3)
If X=0 Then
  MsgBox("Message level is set to 3 - No messages")
  MsgBox("Error. Message level not set.")
End If
End Sub
```

HypSetLinkMacro

Description

HypSetLinkMacro() sets the macro name to be run to perform the dynamic link query action.

Note:

Once the link action is triggered from the Visualize in Excel menu item, the macro name set by this function name will be run.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234.

Syntax

HypSetLinkMacro (vtMacroName)

ByVal vtMacroName As Variant

Parameters

vtMacroName: The name of the macro to be run.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypSetLinkMacro Lib "HsAddin.dll" (ByVal vtMacroName As
Variant) As Long
Sub Auto_Open()
  Sts = HypUseLinkMacro(True)
  Sts = HypSetLinkMacro("Sheet1.Macro8")
End Sub
```

HypSetMenu

Description

HypSetMenu() removes or restores the Hyperion menu from Excel.

Syntax

HypSetMenu(bSetMenu)

ByVal bSetMenu As Boolean

Parameters

bSetMenu: Boolean value indicating whether to remove or restore the Hyperion menu for Excel. A True value indicates that the menu should be restored. A False value indicates that the menu should be removed.

Return Value

Returns 0 if successful. If the menu cannot be set, returns an error code.

Example

```
Declare Function HypSetMenu Lib "HsAddin.dll" (ByVal bSetMenu As Boolean)
As Long
Sub SetMyMenu()
   X=HypSetMenu(TRUE)
End Sub
```

HypSetOverrideFlag

Description

This function sets the override flag value from Hyperion Options. The value of this flag indicates which Provider Services URL will be used. Possible values for bOverride:

- True the current user's Provider Services URL will be used to try to connect to Provider Services
- False the global Provider Services URL will be used to try to connect to Provider Services

Syntax

HypSetOverrideFlag(bOverride)

ByVal vtOverride As Boolean

Parameters

bOverride

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

Declare Function HypSetOverrideFlag Lib "HsAddin.dll" (ByVal vtOverride As Boolean) As Long

```
Sub SubHypSetOverrideTest()
        Dim 1Ret As Long
        'lRet = HypSetOverrideFlag(True)
        lRet = HypSetOverrideFlag(False)
End Sub
```

HypSetPOV

Description

HypSetPOV() sets the POV for the selected sheet.

Syntax

```
HypSetPOV(vtSheetName ParamArray MemberList())
ByVal vtSheetName As Variant
ParamArray MemberList() As Variant
```

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

MemberList: A list of strings which describe the member combination for which a data value will be retrieved. If MemberList is Null or Empty, the top level value is used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypSetPOV Lib "HsAddin" (ByVal vtSheetName, ParamArray
MemberList() As Variant) As Long
Sub SetPOV()
   X=HypSetPOV ("[Book2.xls]Sheet1", "Year#Qtr1", "Market#East")
End Sub
```

HypSetPOVItems

Description

HypSetPOVItems() sets the POV dimensions for the dynamic link query.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

```
HypSetPOVItems (ppMemberList())
```

ParamArray ppMemberList() As Variant

Parameters

ppMemberList: The list of desired POV items in the form Dimension#Current Member.

Return Value

Returns 0 if successful; otherwise, returns the negative error code.

Example

```
Declare Function HypSetLinkMacro Lib "HsAddin.dll" (ByVal vtMacroName As
Variant) As Long

Sub Macro()
   Dim vtGrid as Variant
   Sts = HypConnect("Sheet1", "system", "password", "MyDemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypSetPOVItems ("Scenario#Scenario", "Measures#Measures")
End sub
```

HypSetRowItems

Description

Sets the members for the nth row dimension for this dynamic link query. If the nth row does not exist, a new row is appended.

Note:

This function is used specifically with dynamic link views, as described in "Dynamic Link Views" on page 234

Syntax

HypSetRowItems (vtRowID, vtDimensionName, ppMemberList())

ByVal vtRowID As Variant

ByVal vtDimensionName As Variant

ParamArray ppMemberList() As Variant

Parameters

vtRowID:The row number n.

vtDimensionName: The dimension name.

ppMemberList: The list of member names.

Return Value

Long. Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypSetRowItems Lib "HsAddin.dll" (ByVal vtRowID As
Variant, ByVal vtDimensionName As Variant, ParamArray MemberList() As
Variant) As Long
Sub Macro()
   Dim vtGrid as Variant
   Sts = HypConnect("Sheet1", "system", "password", "DemoBasic")
   Sts = HypRetrieve("Sheet1")
   Range ("B2").Select
   Sts = HypGetSourceGrid ("Sheet1", vtGrid)
   Sts = HypSetRowItems(1, "Product", "100", "200", "300", "400", "Diet",
"Product")
End sub
```

HypSetSheetOption

Description

HypSetSheetOption() sets individual spreadsheet options.

Note:

You can set only one option at a time (this function is not plural).

Syntax

HypSetSheetOption(vtSheetName, vtItem, vtOption)

ByVal vtSheetName As Variant

ByVal vtItem As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtItem: Number indicating which option is to be set. See Table 18 for a list of values.

vtOption: A Boolean value denoting the new value of item.

Table 18 indicates which options are set for which number and the expected data type.

Table 18 vtltems

| vtItem | Option | Data Type and Values |
|--------|---|----------------------|
| 1 | Set zoom in level: | Number |
| | 0 Next level | |
| | 1 All levels | |
| | 2 Bottom level | |
| 2 | Enable Include Selection setting | Boolean |
| 3 | Enable Within Selection Group setting | Boolean |
| 4 | Enable Remove Unselected Groups setting | Boolean |
| 5 | Specify Indent setting: | Number |
| | 0 No indentation | |
| | 1 Indent sub items | |
| | 2 Indent totals | |
| 6 | Enable suppress missing setting | Boolean |
| 7 | Enable suppress zeros setting | Boolean |
| 8 | Enable suppress underscores setting | Boolean |
| 9 | Enable No Access setting | Boolean |
| 10 | Enable Repeated Members setting | Boolean |
| 11 | Enable invalid setting | Boolean |
| 12 | Ancestor Position: | Number |
| | 0 Тор | |
| | 1 Bottom | |
| 13 | Specify Missing text label | Text |
| 14 | Specify No Access label | Text |

| vtItem | Option | Data Type and Values |
|--------|---|----------------------|
| 15 | Cell status: 0 Data 1 Calculation status 2 Process Management | Number |
| 16 | Member Name Display options: 0 Name only 1 Name and Description 2 Description only | Number |

Return Values

Returns 0 if successful. A negative number indicates a local failure. A return value greater than zero indicates a failure on the server.

Example

```
Declare Function HypSetSheetOption Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtItem As Variant, ByVal vtOption As Variant) As Long
Sub SetSheet()
X=HypSetSheetOption(Null, 6, FALSE)
If X=0 Then
  MsgBox("#Missing values will appear. ")
  MsgBox("Error. #Missing option not set.")
End If
End Sub
```

HypSetSubstitutionVariable

Description

HypSetSubstitutionVariable() creates substitution variables on Analytic Server. If the variable already exists, then its value is set to the new specified value.

Syntax

HypSetSubstitutionVariable (vtSheetName, vtApplicationName, vtDatabaseName, vtVariableName, vtVariableValue)

ByVal vtSheetName As Variant

ByVal vtApplicationName As Variant

ByVal vtDatabaseName As Variant

ByVal vtVariableName As Variant

ByVal vtVariableValue As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtApplicationName: The application name to define the scope for the new variable. If vtApplicationName is Null or Empty, the scope of the variable created is global.

vtDatabaseName: The database name to define the scope for the new variable. If vtDatabaseName is Null or Empty, the scope of the variable created is global within the application specified.

vtVariableName: The variable name to be created. Required.

vtVariableValue: The value to be assigned to the variable. Required.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypSetSubstitutionVariable Lib "HsAddin.dll" (ByVal
vtSheetName As Variant, ByVal vtApplicationName As Variant, ByVal
vtDatabaseName As Variant, ByVal vtVariableName As Variant, ByVal
vtVariableValue As Variant) As Long
```

```
Sub Sample_HypSetSubstitutionVariable
  Dim X as Long
  X = HypSetSubstitutionVariable ("Sheet1", "Sample", "Basic",
   "Account", "100")
End Sub
```

HypSubmitData

Description

HypSubmitData() updates the database with modified data from the specified spreadsheet.

Note:

HypSubmitData() is not supported with aggregate storage databases or in a clustered environment.

Note:

The ability to update the database depends on the access permissions of the submitter. To update data, you must have at least Write access to the database.

Syntax

HypSubmitData(vtSheetName)

ByVal vtSheetName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypSubmitData Lib "HsAddin" (ByVal vtSheetName As Variant)
As Long
Worksheets("Sheet1").range("B2").value = 8023
Worksheets("Sheet1").range("B2").Select
sts = HypSubmitData("Sheet1")
```

HypTranslate

Description

HypTranslate() calls the Translate method for Financial Management data sources.

Syntax

```
HypTranslate (vtSheetName, vtRange)
ByVal vtSheetName As Variant
By Val vtRange As Variant
```

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtRange: Range object which refers to the data to be used. Passing an empty or null parameter uses the current selection from the sheet.

Return Value

Returns 0 if successful; otherwise, returns the corresponding error code.

Example

```
Declare Function HypTranslate Lib "HsAddin" (ByVal vtSheetName As Variant,
ByVal vtRange As Variant) As Long
sts = HypTranslate ("Sheet1", Empty)
```

HypUndo

Description

HypUndo() restores the previous database view. A database view is the view of the spreadsheet after performing Zoom In, Zoom Out, Keep Only, Remove Only, or Refresh commands.

Syntax

HypUndo (vtSheetName)

ByVal vtSheetName As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypUndo Lib "HsAddin.dll" (ByVal vtSheetName As Variant)
As Long
Sub Undo()
  X=HypUndo("[Book2.xls]Sheet1")
End Sub
```

HypUseLinkMacro

Description

HypUseLinkMacro() is used to set the flag to specify the type of link view, static or dynamic.

Note:

Static and dynamic link views share the same menu option; therefore, it is necessary to turn the flag on before performing the dynamic link query. Once done with dynamic link views, turn the flag off.

Note:

This function is used specifically with link views, as described in "Dynamic Link Views" on page 234.

Syntax

HypUseLinkMacro (bSetView)

ByVal bSetView as Boolean

Parameters

bSetView: When flag is set to true, dynamic link is performed. When the flag is set to false, static link is performed.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypUseLinkMacro Lib "HsAddin.dll" (ByVal bUse As Boolean)
As Long
Sub Macro()
   Sts = HypUseLinkMacro(True)
End sub
```

HypZoomIn

Description

HypZoomIn() retrieves and expands data from Smart View based on the selected members.

Syntax

HypZoomIn(vtSheetName, vtSelection, vtLevel, vtAcross)

ByVal vtSheetName As Variant

ByVal vtSelection As Variant

ByVal vtLevel As Variant

ByVal vtAcross As Variant (not used)

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtSelection: Range object which refers to the members that will be zoomed. If selection is Null or Empty, the active cell is used

vtLevel: Number indicating the granularity of the zoom. The following table describes the valid level numbers and their actions:

- 1 Next level
- 2 All levels
- 3 Bottom level

If level is Null or Empty, the default is used.

vtAcross: Not used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypZoomIn Lib "HsAddin.dll" (ByVal sheetName As Variant,
ByVal vtSelection As Variant, ByVal vtLevel As Variant, ByVal vtAcross As
Variant) As Long
Sub ZoomData()
X=HypZoomIn("[Book2.xls]Sheet1", RANGE("B3"), 1, FALSE)
If X = 0 Then
   MsgBox("Zoom successful.")
Else
   MsgBox("Zoom failed.")
End If
End Sub
```

HypZoomOut

Description

HypZoomOut() collapses the view of data based on the selected members.

Syntax

HypZoomOut(vtSheetName, vtSelection) ByVal vtSheetName As Variant ByVal vtSelection As Variant

Parameters

vtSheetName: Text name of worksheet to operate on. vtSheetName is of the form "[Book.xls] Sheet". If vtSheetName is null, empty, or invalid (nonexistent or name does not match exactly), the active sheet is used.

vtSelection: Range object which refers to the members that will be zoomed out. If selection is Null or Empty, the active cell is used.

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypZoomOut Lib "HsAddin.dll" (ByVal vtSheetName As
Variant, ByVal vtSelection As Variant) As Long
Sub UnZoomData()
X=HypZoomOut("[Book2.xls]Sheet1", RANGE("B3"))
If X = 0 Then
   MsgBox("Zoom out successful.")
Else
   MsgBox("Zoom out failed.")
End If
End Sub
```

About Visual Basic Menu Equivalent Functions

These Visual Basic functions are identical to the equivalent commands on the Hyperion menu. Use the functions to perform actions as if you selected them from the menu. The requirements for the functions are the same as those for the menu commands. For example, if you must be logged in to an instance of Analytic Server to use a menu command, you must also be logged in to an instance of Analytic Server to use the equivalent Visual Basic function.

Visual Basic Menu Functions

Table 19 lists the Smart View VBA menu equivalent functions alphabetically and specifies the data source provider that supports the function. A detailed description for each function, including the syntax, parameters, return value, and sample code, follows the table.

Table 19 VBA Menu Equivalent Functions and Supported Providers

| VBA Functions | Data Source Types | |
|---------------------|--|--|
| HypMenuVCalculation | Essbase | |
| HypMenuVConnect | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVDisconnect | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVKeepOnly | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVOptions | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVPivot | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVRemoveOnly | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVRefresh | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVRetrieve | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVSubmitData | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVZoomIn | Essbase, Financial Management, Hyperion Enterprise | |
| HypMenuVZoomOut | Oracle Essbase, Financial Management, Oracle's Hyperion® Enterprise® | |

HypMenuVCalculation

Description

HypMenuVCalculation() can be used to open the Calculation Scripts dialog box and calculate the active database or checks on the status of an active database calculation.

Syntax

HypMenuVCalculation()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

```
Declare Function HypMenuVCalculation Lib "HsAddin.dll"() As Long
Sub MCalc()
  X=HypMenuVCalculation()
End Sub
```

HypMenuVConnect

Description

HypMenuVConnect() can be used to connect to a data source instance.

Syntax

HypMenuVConnect()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypMenuVConnect Lib "HsAddin.dll"() As Long
Sub MConn()
  X=HypMenuVConnect()
End Sub
```

HypMenuVDisconnect

Description

HypMenuVDisconnect() disconnects you from any currently connected databases.

Syntax

HypMenuVDisconnect()

Return Value

```
Declare Function HypMenuVDisconnect Lib "HsAddin.dll"() As Long
Sub MDisConn()
  X=HypMenuVDisconnect()
End Sub
```

HypMenuVKeepOnly

Description

HypMenuVKeepOnly() retains only the selected member (the active cell) or member range in the sheet.

Syntax

HypMenuVKeepOnly()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypMenuVKeepOnly Lib "HsAddin.dll"() As Long
Sub MKeepOnly()
   X=HypMenuVKeepOnly()
End Sub
```

HypMenuVOptions

Description

HypMenuVOptions() enables you to select options for the active sheet and customize the behavior of Smart View, using the Options dialog box.

Syntax

HypMenuVOptions()

Return Value

```
Declare Function HypMenuVOptions Lib "HsAddin.dll"() As Long
Sub MOptions()
  X=HypMenuVOptions()
End Sub
```

HypMenuVPivot

Description

HypMenuVPivot() changes the orientation (from row to column or from column to row) of the group of members associated with the active cell.

Syntax

HypMenuVPivot()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypMenuVPivot Lib "HsAddin.dll"() As Long
Sub MPivot()
  X=HypMenuVPivot()
End Sub
```

HypMenuVRemoveOnly

Description

HypMenuVRemoveOnly() removes only the selected member (the active cell) or member range in the sheet.

Syntax

HypMenuVRemoveOnly()

Return Value

```
Declare Function HypMenuVRemoveOnly Lib "HsAddin.dll"() As Long
Sub MRemoveOnly()
  X=HypMenuVRemoveOnly()
End Sub
```

HypMenuVRefresh

Description

HypMenuVRefresh() retrieves data into the active sheet, and places the data at the beginning of the active worksheet.

Syntax

HypMenuVRefresh()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypMenuVRefresh Lib "HsAddin.dll"() As Long
Sub MRetrieve()
  X=HypMenuVRefresh()
End Sub
```

HypMenuVRetrieve

Description

HypMenuVRetrieve() retrieves data from the database.

Syntax

HypMenuVRetrieve()

Return Value

```
Declare Function HypMenuVRetrieve Lib "HsAddin.dll"() As Long
Sub MRetrieve()
  X=HypMenuVRetrieve()
End Sub
```

HypMenuVSubmitData

Description

HypMenuVSubmitData() updates the active database on the server with data that has been modified in your sheet or marked as "dirty" using the SetCellsDirty call.

Syntax

HypMenuVSubmitData()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypMenuVSubmitData Lib "HsAddin.dll"() As Long
Sub MSubmit()
   X=HypMenuVSubmitData()
End Sub
```

HypMenuVZoomIn

Description

HypMenuVZoomIn() retrieves and expands data from the data source according to the options specified in the Options dialog box.

Syntax

HypMenuVZoomIn()

Return Value

```
Declare Function HypMenuVZoomIn Lib "HsAddin.dll"() As Long
Sub MZoomIn()
  X=HypMenuVZoomIn()
End Sub
```

HypMenuVZoomOut

Description

HypMenuVZoomOut() collapses the view of data according to the options specified in the Options dialog box.

Syntax

HypMenuVZoomOut()

Return Value

Returns 0 if successful; otherwise, returns the appropriate error code.

Example

```
Declare Function HypMenuVZoomOut Lib "HsAddin.dll"() As Long
Sub MZoomOut()
  X=HypMenuVZoomOut()
End Sub
```

Error Codes

Table 20 lists return error codes for VBA.

Table 20 Error Codes

| Error Code | Number |
|------------------|--------|
| SS_OK | 0L |
| SS_INIT_ERR | -1L |
| SS_TERM_ER | -2L |
| SS_NOT_INIT | -3L |
| SS_NOT_CONNECTED | -4L |
| SS_NOT_LOCKED | -5L |

| Error Code | Number |
|------------------------|--------|
| SS_INVALID_SSTABLE | -6L |
| SS_INVALID_SSDATA | -7L |
| SS_NOUNDO_INFO | -8L |
| SS_CANCELED | -9L |
| SS_GLOBALOPTS | -10L |
| SS_SHEETOPTS | -11L |
| SS_NOTENABLED | -12L |
| SS_NO_MEMORY | -13L |
| SS_DIALOG_ERROR | -14L |
| SS_INVALID_PARAM | -15L |
| SS_CALCULATING | -16L |
| SS_SQL_IN_PROGRESS | -17L |
| SS_FORMULAPRESERVE | -18L |
| SS_INTERNALSSERROR | -19L |
| SS_INVALID_SHEET | -20L |
| SS_NOACTIVESHEET | -21L |
| SS_NOTCALCULATING | -22L |
| SS_INVALIDSELECTION | -23L |
| SS_INVALIDTOKEN | -24L |
| SS_CASCADENOTALLOWED | -25L |
| SS_NOMACROS | -26L |
| SS_NOREADONLYMACROS | -27L |
| SS_READONLYSS | -28L |
| SS_NOSQLACCESS | -29L |
| SS_MENUALREADYREMOVED | -30L |
| SS_MENUALREADYADDED | -31L |
| SS_NOSPREADSHEETACCESS | -32L |
| SS_NOHANDLES | -33L |

| Error Code | Number |
|--|--------|
| SS_NOPREVCONNECTION | -34L |
| SS_LROERROR | -35L |
| SS_LROWINAPPACCESSERR | -36L |
| SS_DATANAVINITERR | -37L |
| SS_PARAMSETNOTALLOWED | -38L |
| SS_SHEET_PROTECTED | -39L |
| SS_CALCSCRIPT_NOTFOUND | -40L |
| SS_NOSUPPORT_PROVIDER | -41L |
| SS_INVALID_ALIAS | -42L |
| SS_CONN_NOT_FOUND | -43L |
| SS_APS_CONN_NOT_FOUND | -44L |
| SS_APS_NOT_CONNECTED | -45L |
| SS_APS_CANT_CONNECT | -46L |
| SS_CONN_ALREADY_EXISTS | -47L |
| SS_APS_URL_NOT_SAVED | -48L |
| SS_MIGRATION_OF_CONN_NOT_ALLOWED | -49L |
| SS_CONN_MGR_NOT_INITIALIZED | -50L |
| SS_FAILED_TO_GET_APS_OVERRIDE_PROPERTY | -51L |
| SS_FAILED_TO_SET_APS_OVERRIDE_PROPERTY | -52L |
| SS_FAILED_TO_GET_APS_URL | -53L |
| SS_APS_DISCONNECT_FAILED | -54L |
| SS_OPERATION_FAILED | -55L |



Accessibility

In This Appendix

| Enabling Accessibility for Smart View | .333 |
|---------------------------------------|------|
| Keyboard Equivalents | .333 |
| Accessibility in Office 2007 | .339 |

This appendix describes the accessibility features of Smart View. For information regarding supported assistive technologies, refer to the Oracle Hyperion Enterprise Performance Management System Installation Start Here.

Enabling Accessibility for Smart View

You do not need to enable accessibility specifically for Smart View; it is always in accessible mode. Smart View output is in the form of Excel spreadsheets, Word documents, and PowerPoint slides, which are accessible through Microsoft Office. For information about Excel, Word, or PowerPoint accessibility, refer to Microsoft Office product documentation.

Keyboard Equivalents

This section describes the keyboard equivalents for the Hyperion menu, The Common Provider Connections interface, the Independent Provider Connections interface, and the Report Designer.

Hyperion Menu Keyboard Equivalents

Table 21 lists the keyboard equivalents for all Hyperion menu items and submenu items if you use Office 2003. If you use Office 2007, refer to "Accessibility in Office 2007" on page 339.

Table 21 Hyperion Menu Keyboard Equivalents

| Key | Menu Item |
|---------|---------------------|
| Alt+Y+G | Data Source Manager |
| Alt+Y+T | Activate |

| Key | Menu Item |
|------------------------------|---|
| Alt+Y+E | Reset Connection |
| Alt +Y +Y | Ad Hoc Analysis |
| Submenu for Ad Hoc Analysis: | |
| Alt+Y+Y+Z | Zoom In |
| Alt+Y+Y+U | Zoom out |
| Alt+Y+Y+V | Pivot |
| Alt+Y+Y+T | Pivot to POV |
| Alt+Y+Y+K | Keep Only |
| Alt+Y+Y+O | Remove Only |
| Alt+Y+Y+D | Drill Through Reports |
| Alt+Y+Y+Q | View Qualified Member Name |
| Alt+Y+Y+P | Capture Formatting |
| Alt+Y+Y+C | Change Alias Table |
| Alt +Y + F | Forms |
| Sub Menu for Forms: | |
| Alt+Y+F+F | Select Forms |
| Alt+Y+F+I | Instructions |
| Alt+Y+F+M | Add Member |
| Alt+Y+F+T | Take Offline |
| Alt+Y+F+Y | Sync Back to Server |
| Alt+Y+F+R | Refresh Offline Definition |
| Alt+Y+F+L | Lock |
| Alt+Y+F+X | Expand |
| Alt+Y+F+P | Collapse |
| Alt +Y+U | Functions |
| Sub Menu for Functions: | |
| Alt+Y+U+N | Function Builder |
| Alt+Y+U+G | Migrate Active Workbook (Oracle Hyperion Financial Management, Fusion Edition) |

| Кеу | Menu Item |
|---|---|
| Alt+Y+U+B | Migrate Batch (Financial Management) |
| Alt+Y+U+E | Migrate Batch (Hyperion Enterprise) |
| Alt+Y+U+R | Migrate Active Workbook (Hyperion Enterprise) |
| Alt+Y+U+I | Insert Connection List |
| Alt +Y + O | Reporting and Analysis Document |
| Sub Menu for Oracle's Hyperion Reporting and Analysis Document: | |
| Alt+Y+L+I | Import |
| Alt+Y+L+E | Edit |
| Alt +Y + C | Copy Data Points |
| Alt +Y + V | Paste Data Points |
| Alt +Y + M | POV Manager |
| Alt +Y + N | Undo |
| Alt +Y + D | Redo |
| Alt +Y + I | Member Selection |
| Alt +Y + R | Refresh |
| Alt +Y + A | Refresh All |
| Alt +Y + D | Data Perspective |
| Alt +Y + B | Submit Data |
| Alt +Y + L | Calculation Options |
| Sub Menu for Calculations Options: | |
| Alt+Y+L+C | Calculate |
| Alt+Y+L+F | Force Calculate |
| Alt+Y+L+A | Calculate Contribution |
| Alt+Y+L+O | Force Calculate Contribution |
| Alt+Y+L+T | Translate |
| Alt+Y+L+R | Force Translate |
| Alt+Y+L+N | Consolidate |
| Alt+Y+L+S | Consolidate All |

| Key | Menu Item |
|------------------------|---------------------------|
| Alt+Y+L+L | Consolidate All with Data |
| Alt+Y+L+B | Business Rules |
| Alt+Y+L+U | Rules on Form |
| Alt +Y + J | Adjust |
| Alt +Y + X | Cell Text |
| Alt +Y + P | Supporting Details |
| Alt +Y + D | Document Link |
| Alt +Y + S | Options |
| Alt +Y + Q | Query |
| Submenu for Query: | |
| Alt+Y+Q+Q | Query Designer |
| Alt+Y+Q+R | Run Report |
| Alt+Y+Q+E | Execute MDX |
| Alt+Y+Q+D | Data Filter |
| Alt +Y+W | Link view |
| Submenu for Link View: | |
| Alt+Y+W+H | Visualize in HVE |
| Alt+Y+W+E | Visualize in Excel |
| Alt +Y+H | Help |
| Alt +Y +Z | About |

Independent Provider Connections Keyboard Equivalents

Table 22 lists the keyboard equivalents functions and navigation for the Independent Provider Connections interface of the Data Source Manager. After invoking a function in the Data Source Manager, focus moves to the grid.

 Table 22
 Independent Provider Connections Keyboard Equivalents

| Key | Action |
|---------|---|
| Alt+Y+G | In Office 2003, open the Data SourceManager; in Office 2007, move back and forth between grid and Data Source Manager |

| Кеу | Action |
|---------------------------|--|
| Alt+F4 | (Office 2003) Close the Data Source Manager and move to grid |
| Alt+S | Switch between Common Provider Connections, Independent Provider Connections, and Simulation and Forecasting Workbooks |
| Alt+C | Connect |
| Alt+D | Disconnect |
| Alt+T | Set as Default |
| Alt+P | Change Password |
| Alt+A | Add |
| Alt+E | Edit |
| Alt+R | Delete |
| Tab | Switch between toolbar and list |
| Shift+F10 | View properties menu for a connection |
| Up and down arrow keys | In list of connections—move through items in list |
| | In toolbar—move through items in popup menus |
| Left and right arrow keys | In toolbar, move through buttons |
| Spacebar | Select a toolbar button |

Common Provider Connections Keyboard Equivalents

Table 23 lists the keyboard equivalents of Common Provider Connections functions.

 Table 23
 Common Provider Connections Keyboard Equivalents

| Key | Action |
|-------|--|
| Alt+P | Connect to Provider Services |
| Alt+S | Switch between Common Provider Connections, Independent Provider Connections, and Simulation and Forecasting Workbooks |
| Alt+V | Views |
| Alt+O | Add to Views |
| Alt+T | Oracle Hyperion Provider Services Properties |
| Alt+X | Expand |
| Alt+L | Collapse |

| Key | Action |
|---------------------------|--|
| Alt+C | Add |
| Alt+E | Modify |
| Alt+D | Delete |
| Alt+Y | Sync with Views |
| Alt+B | Smart Slice operations |
| Alt+I | Insert into Reports |
| Alt+H | Cascade Ad Hoc Grid |
| Tab | Switch between toolbar and tree list |
| Left and right arrow keys | In toolbar — move through buttons |
| | In tree view—expand or collapse a node |
| Spacebar | Select a toolbar button |
| Alt+F4 | (Office 2003) Close the Data Source Manager and move to grid |
| Shift+F10 | View properties menu for a connection |
| Up and down arrow keys | In tree view—move through items in list |
| | In toolbar—move through items in popup menus |
| Escape+Tab | Switch between Data Source Manager and Report Designer (if Report Designer is displayed) |

Report Designer Keyboard Equivalents

Table 24 lists the keyboard equivalents of Report Designer functions and navigation.

 Table 24
 Report Designer Keyboard Equivalents

| Key | Action |
|-------|--------------------------------------|
| Alt+V | Views |
| Alt+X | Expand |
| Alt+L | Collapse |
| Alt+I | Insert reporting objects |
| Alt+D | Delete |
| Alt+H | Cascade report across workbook |
| Tab | Switch between toolbar and tree list |

| Key | Action |
|---------------------------|--|
| Left and right arrow keys | In toolbar — move through buttons |
| | In tree view—expand or collapse a node |
| Spacebar | Select a toolbar button |
| Alt+F4 | (Office 2003) Close the Data Source Manager and move to grid |
| Shift+F10 | View properties menu for a reporting object |
| Up and down arrow keys | Move through items in the tree view |
| Escape+Tab | Switch between Data Source Manager and Report Designer (if Report Designer is displayed) |

Accessibility in Office 2007

If you use Office 2007 with Smart View, you can move through the Excel grid and all Smart View components using the F6 key.

In Office 2007, most menu items listed in Table 21 are displayed in the Hyperion ribbon, and you can use the same keyboard equivalents for them. Other items are available from the 2007 Hyperion menu located on the Add-in ribbon. Items on this Hyperion menu are accessed using the commands listed in Table 21, but substituting Alt+X for Alt. For Example, you would use Alt+X+Y+F+M instead of Alt+Y+F+M for the Add Member command.

Table 25 lists keyboard equivalents for using the Hyperion ribbon, Add-in ribbon, and the 2007 Hyperion menu

Table 25 Hyperion Ribbon Keyboard Equivalents

| Key | Action |
|---------|--|
| Alt+Y | Open the Hyperion ribbon |
| Alt+X | Open the Add-in ribbon |
| Alt+X+Y | Open the Hyperion menu from the Add-in ribbon, where you can access Oracle Hyperion Smart View for Office, Fusion Edition items that are not displayed on the Hyperion ribbon. |

Index

| Symbols | enabling double-click for Ad Hoc operations, 209 |
|---|---|
| #Missing label | indenting member names, 208 |
| handling in free-form, 124 | navigating the spreadsheet without retrieving data, |
| #Missing rows, suppressing, 208 | 209 |
| #Missing Values in data forms, 65 | retaining members, 211 |
| #NoData label, replacing, 213 | suppressing types of data, 207 |
| | Undo and Redo, enabling, 209, 210 |
| | Undo and Redo, using, 210 |
| A | zoom options, 210 |
| accessibility, 333 | Ad Hoc tab |
| keyboard equivalents, 333 | ancestor position in hierarchies, 211 |
| using Smart View with Office 2007, 339 | double-click for Ad Hoc operations, enabling, 209 |
| accessibiliyt | indenting member names in spreadsheets, 208 |
| enabling, 333 | member retention options, 211 |
| accessing a drill-through report, 115 | navigating the spreadsheet without retrieving data, |
| Ad Hoc accessing Hybrid Analysis data, 108 | 209 |
| | suppressing types of data, 207 |
| adding cell text, 110 calculating data, 111 | Undo and Redo, enabling, 209, 210 |
| consolidating data, 111 | Undo and Redo, using, 210 |
| displaying aliases for member names, 107 | zoom options, 210 |
| Dynamic Time Series members, 102 | Add button, 18 |
| enabling double-clicking for, 209 | Adjust button, 17 |
| free-form, 119 | adjusting values, 73 |
| | aliases, displaying for member names, 107 |
| keeping data, 109 pivot, 108 | All Levels zoom option, 210 |
| • | ancestor position in hierarchies, 211 |
| removing data, 110 selecting members, 97 | asymmetric reports, 156 |
| submitting data, 114 | definition of, 156 |
| translating data, 113 | retrieving data into, 156 |
| viewing cell text, 111 | attribute dimensions and members, 98 |
| working with formulas, 114 | attribute dimensions, in free-form grids, 122 |
| zoom in, 104 | automatic deployment, 29 |
| zoom out, 105 | |
| ad hoc and data forms, 238 | В |
| Ad Hoc features, 21 | background POV, 222 |
| ad hoc options, 207 | base currency, overriding, 67 |
| ancestor position in hierarchies, 211 | benefits, 15 |
| and the position in includence, 211 | |

| Bottom Level zoom option, 210 | C |
|--------------------------------|---|
| business rules | Calculate Currencies business rule, 72 |
| Calculate Currencies, 72 | Calculate Form business rule, 72 |
| Calculate Form, 72 | calculating data in Ad Hoc grids, 111 |
| launching, 69 | calculating rows, inserting, 112 |
| overview, 69 | calculation scripts See business rules |
| runtime prompts in, 70 | Calculation Status grid display option, 213 |
| Business Rules button, 17 | cascading, 55 |
| buttons | ad hoc grids, 55 |
| Add, 18 | Smart Slice reports, 55 |
| Adjust, 17 | cell status, 213 |
| Business Rules, 17 | cell styles options |
| Cell Text, 17 | order of precedence of styles, 218 |
| Collapse, 17 | Cell Styles tab |
| Copy Data Points, 17 | order of precedence of styles, 218 |
| Data Source Manager, 16 | Cell Text button, 17 |
| Expand, 17 | cell text in Ad Hoc grids, 110 |
| Function Builder, 17 | cell text in data forms |
| Instructions, 17 | adding, 65 |
| Keep Only, 16 | viewing and editing, 66 |
| Lock, 17 | cells, 64. See also data |
| Member Selection, 17 | copying and pasting in data forms, 64 |
| Options, 18 | currency code, location of, 68 |
| Paste Data Points, 17 | currency, changing, 67 |
| Pivot, 16 | navigating in data forms, 59 |
| POV Manager, 17 | selecting a range in data forms, 63 |
| Query Designer, 17 | chart |
| Redo, 17 | Report Designer, 53 |
| Refresh, 16 | Collapse button, 17 |
| Refresh All, 16 | comments in free-form |
| Refresh Offline Definition, 18 | handling, 124 |
| Remove, 18 | preserving, 124 |
| Remove Only, 16 | common provider |
| Rules on Form, 17 | associating connection to worksheet, 33 |
| Run Reports, 18 | connecting to data sources, 32 |
| Select Form, 17 | disconnecting, 33 |
| Submit Data, 16 | Common Provider Connections |
| Supporting Details, 17 | keyboard equivalents, 337 |
| Sync Back, 17 | common provider connections, 31, 32 |
| Take Offline, 17 | compatibility |
| Undo, 16 | data sources, 29 |
| Visualize in Excel, 18 | concurrent use of Smart View and Essbase Spreadshee |
| Visualize in HVE, 18 | Add-in, 28 |
| Zoom In, 16 | Connecting to data sources |
| Zoom Out, 16 | from common provider connections, 32 |
| • | Essbase connections |
| | migrating to Data Source Manager, 41 |

| consolidating data in Ad Hoc grids, 113 | data source |
|---|--|
| Copy Data Points button, 17 | adding a new connection, 36 |
| copying and pasting data points into Word and | adding from Shared Services, 38 |
| PowerPoint, 159 | adding through direct connection, 36 |
| copying data in data forms, 64 | associating with a worksheet, 39 |
| creating | checking for active connections, 41 |
| an offline connection, 200 | connecting to a, 39 |
| Crystal BallCrystal Ball, 42 | deleting, 41 |
| currency | disconnecting from, 41 |
| changing for a data cell, 67 | editing an existing connection, 39 |
| currency codes | resetting connections, 41 |
| location of, 68 | data source connections, 31 |
| custom formats, 63 | Data Source Manager |
| , | adding and deleting connections, 34 |
| _ | and Connections, 31 |
| D | views, 33 |
| data | Data Source Manager button, 16 |
| adjusting, 73 | Decimal Places grid display option, 215 |
| entering percentage values, 64 | decimal separator in data forms, 63 |
| missing values, 65 | default connection |
| submitting, 68 | setting, 40 |
| subtotaling, 67 | default connection, setting, 40 |
| data display options in the grid | dirty cells |
| Calculation Status option, 213 | defined, 123 |
| Data option, 213 | submitting in free-form, 123 |
| Decimal Places option, 215 | disabling Smart View, 28 |
| Member Name Only option, 214 | Disconnecting |
| Process Management option, 213 | common provider, 33 |
| Scale options, 215 | display options, 212 |
| data filtering, 91 | Excel formatting, using, 215 |
| data form features, 24 | messages, specifying types to display, 217 |
| data forms | specifying UI colors, 214 |
| ad hoc, 238 | Display tab |
| adding members, 62 | Excel formatting, 215 |
| currency, changing, 67 | messages, specifying types to display, 217 |
| customizing format, 63 | specifying UI colors, 214 |
| linked forms, 62 | displaying aliases for member names, 107 |
| navigating, 59 | Reporting and Analysisdocument import features, 26 |
| opening, 59 | double-clicking for Ad Hoc operations, enabling, |
| refreshing offline, 203 | 209 |
| selecting multiple, 59 | drill-through reports |
| taking offline, 199, 200 | accessing, 115 |
| viewing instructions, 60 | working with, 115 |
| viewing multiple, 60 | 5 |
| working with formulas, 83 | duplicate member names, resolving in free-form |
| working with offline, 201 | mode, 138 |
| Data grid display option, 213 | dynamic data access features, 24 |
| data perspective, 115 | dynamic data points, 159 |

| dynamic link views | resolving member names, 138 |
|---|--|
| working with, 234 | retrieving, 135 |
| Dynamic Time Series members | submitting dirty cells, 123 |
| in Ad Hoc grids, 102 | valid grids, 125 |
| | Function Builder, 185 |
| E | Function Builder button, 17 |
| editing documents, 166 | function features, 25 |
| enabling Smart View, 28 | function grid |
| Excel formatting, using, 215 | Report Designer, 52 |
| Expand button, 17 | functions |
| Expand button, 17 | creating manually, 196 |
| | editing, 197 |
| F | error codes, 198 |
| File size | overview, 184 |
| reducing, 217 | running, 197 |
| filtering data, 91 | selecting members for, 188 |
| filtering members, 90, 98 | |
| format strings, 214 | G |
| formats in free-form | Google, 18 |
| preserving, 124 | |
| forms, data | graphical views of grids, 163 grid components of free-form, 120 |
| refreshing offline, 203 | guidelines for free-form reporting, 121 |
| taking offline, 199, 200 | guidennes for free-form reporting, 121 |
| using linked forms, 62 | |
| working with offline, 201 | H |
| formulas | HsCurrency function |
| in ad hoc grids, 114 | syntax, 190 |
| in data forms, 83 | using smart tags to retrieve entity currency, 193 |
| preserving when POV changes, 216 | HsDescription function |
| formulas in free-form | syntax, 190 |
| preserving, 124 | using smart tags to display POV description, 194 |
| free-form, 119 | HsGetText function |
| #Missing, handling, 124 | syntax, 191 |
| asymmetric reports, 156 | using smart tags to import cell text, 193 |
| attribute dimensions, working with, 122 | HsGetValue function |
| comment handling, 124 | syntax, 189 |
| comments, preserving, 124 | using smart tags to retrieve single value, 192 |
| constructing a report, 135 | HsLabel function, 191 |
| dirty cells, submitting, 123 | HsSetText function, 191 |
| entering text, 119 | HsSetValue function, 189 |
| example scenario, 139, 145 | Hybrid Analysis, accessing data from a relational |
| formats, preserving, 124 | source, 108 |
| formulas, preserving, 124 | HypCalculate VBA function, 241 |
| grid components, 120 | HypCalculateContribution VBA function, 242 |
| guidelines, 121 | HypCell VBA function, 243 |
| invalid grids, 125 | HypConnect VBA function, 244 |
| preserving comments, formulas, formats, 124 | HypConnected VBA function, 245 |

| HypConnectionExists VBA function, 245 | HypIsParent VBA function, 286 |
|---|--|
| HypConnectToAPS VBA function, 246 | HypIsUDA VBA function, 287 |
| HypConsolidate VBA function, 247 | HypKeepOnly VBA function, 288 |
| HypConsolidateAll VBA function, 247 | HypListCalcScripts VBA function, 289 |
| HypConsolidateAllWithData VBA function, 248 | HypMenuVCalculation VBA function, 324 |
| HypCreateConnection VBA function, 249 | HypMenuVConnect VBA function, 325 |
| HypCreateConnectionEX VBA function, 251 | HypMenuVDisconnect VBA function, 325 |
| HypDeleteCalc VBA function, 252 | HypMenuVKeepOnly VBA function, 326 |
| HypDisconnect VBA function, 253 | HypMenuVOptions VBA function, 326 |
| HypDisconnectEx VBA function, 254 | HypMenuVPivot VBA function, 327 |
| HypDisconnectFromAPS, 255 | HypMenuVRefresh VBA function, 328 |
| HypDisplayToLinkView, 255 | HypMenuVRemoveOnly VBA function, 327 |
| HypDisplayTwoDimArray, 256 | HypMenuVRetrieve VBA function, 328 |
| Hyperion menu | HypMenuVSubmitData VBA function, 329 |
| keyboard equivalents, 333 | HypMenuVZoomIn VBA function, 329 |
| Hyperion Visual Explorer, 163 | HypMenuVZoomOut VBA function, 330 |
| Hyperion Visual Explorer features, 24 | HypMigrateConnectionToDataSourceMgr VBA |
| HypExecuteCalcScript VBA function, 256 | function, 291 |
| HypExecuteMDXEx VBA function, 257 | HypOtlGetMemberInfo VBA function, 291 |
| HypExecuteQuery VBA function, 259 | HypPivot VBA function, 293 |
| HypFindMember VBA function, 260 | HypPivotToGrid VBA function, 293 |
| HypForceCalculate VBA function, 261 | HypPivotToPOV VBA function, 294 |
| HypForceCalculateContribution VBA function, 262 | HypQueryMembers VBA function, 295 |
| HypForceTranslate VBA function, 263 | HypRedo VBA function, 297 |
| HypFreeDataPoint VBA function, 264 | HypRemoveConnection VBA function, 298 |
| HypGetAncestor VBA function, 264 | HypRemoveOnly VBA function, 299 |
| HypGetChildren VBA function, 265 | HypResetFriendlyName VBA function, 300 |
| HypGetColCount VBA function, 266 | HypRetrieve VBA function, 301 |
| HypGetColItems VBA function, 267 | HypRetrieveRange VBA function, 301 |
| HypGetConnectionInfo VBA function, 268 | HypSetActiveConnection VBA function, 302 |
| HypGetCurrentAPSURL VBA function, 270 | HypSetBackgroundPOV VBA function, 304 |
| HypGetDataPoint VBA function, 270 | HypSetCellsDirty VBA function, 304 |
| HypGetGlobalOption VBA function, 271 | HypSetColItems VBA function, 305 |
| HypGetLinkMacro VBA function, 273 | HypSetConnAliasTable VBA function, 306 |
| HypGetOverrideFlag VBA function, 274 | HypSetConnectionInfo VBA function, 307 |
| HypGetParent VBA function, 274 | HypSetCurrentAPSURL VBA function, 308 |
| HypGetPOVCount VBA function, 275 | HypSetGlobalOption VBA function, 309 |
| HypGetPOVItems VBA function, 276 | HypSetLinkMacro VBA function, 310 |
| HypGetRowCount VBA function, 277 | HypSetMenu VBA function, 311 |
| HypGetRowItems VBA function, 277 | HypSetOverrideFlag VBA function, 312 |
| HypGetSheetOption VBA function, 278 | HypSetPOV VBA function, 313 |
| HypGetSourceGrid VBA function, 280 | HypSetPOVItems VBA function, 313 |
| HypGetSubstitutionVariable VBA function, 281 | HypSetRowItems VBA function, 314 |
| HypIsAttribute VBA function, 283 | HypSetSheetOption, 315 |
| HypIsConnectedToAPS VBA function, 284 | HypSetSubstitutionVariable VBA function, 317 |
| HypIsDescendant VBA function, 284 | HypSubmitData VBA function, 318 |
| HypIsExpense VBA function, 285 | HypTranslate VBA function, 319 |

| HypUndo VBA function, 320 | M | |
|---|---|--|
| HypUseLinkMacro VBA function, 321 | MDX queries, 87, 93 | |
| HypZoomIn VBA function, 321 | Member Name Only grid display option, 214 | |
| HypZoomOut VBA function, 323 | member names | |
| , 2 | displaying aliases for, 107 | |
| | entering in spreadsheet, 119 | |
| * 1 | member retention options in spreadsheets, 211 | |
| Indentation options | Member Selection button, 17 | |
| None, 208 | member selection differences, 101 | |
| Subitems, 208 | members | |
| Totals, 208 | taking offline, 200 | |
| indenting member names in spreadsheets, 208 | messages | |
| Independent Provider Connections | specifying type to display, 217 | |
| keyboard equivalents, 336 | migrating | |
| independent provider connections, 31 | Essbaseconnections, 41, 229 | |
| about, 35 | considerations, 227 | |
| tasks you can perform using, 35 | converting a workbook, 228 | |
| inserting | converting multiple workbooks, 228 | |
| calculating rows, 112 | migration utility, 227 | |
| non-calculating rows, 112 | | |
| installing Smart View, 27 | | |
| Instructions button, 17 | N | |
| introduction, 15 | navigating the spreadsheet without retrieving data, | |
| invalid rows, suppressing, 208 | 209 | |
| | new versions of Smart View, 29 | |
| K | Next Level zoom option, 210 | |
| Keep Only button, 16 | No Access rows, suppressing, 208 | |
| Keep Only option, 109 | No Data rows, suppressing, 208 | |
| keyboard equivalents | non-calculating rows, inserting, 112 | |
| Common Provider Connections, 337 | None indentation option, 208 | |
| Smart View functions, 333 | | |
| Hyperion menu functions, 333 | 0 | |
| Independent Provider Connections, 336 | Office 2007 | |
| Report Designer, 338 | accessibility, 339 | |
| | offline. See Offline Planning | |
| | offline connection | |
| 11.1 | creating, 200 | |
| labels, entering in spreadsheet, 119 | offline data forms, 199 | |
| line item detail, 79 | refreshing, 203 | |
| link views. See dynamic link views | taking offline, 200 | |
| linked data forms, 62 | working with, 201 | |
| local currency | Offline Planning | |
| changing, 67 | creating an offline connection, 200 | |
| Lock button, 17 | refreshing data in, 203 | |
| logging | selecting members to take offline, 200 | |
| messages, 217 | taking data forms offline, 200 | |
| | working in, 199, 201 | |
| | | |

| Options button, 18 | creating from default reports, 88 |
|---|--|
| Options dialog box | editing, 90 |
| ad hoc options, 207 | extracting from existing reports, 89 |
| Ad Hoc tab, 207 | filtering members, 90 |
| cell styles options, 218 | Query Designer, 87 |
| Cell Styles tab, 218 | limitations, 87 |
| display options, 212 | refreshing queries, 92 |
| Display tab, 212 | time-related data, 92 |
| order precedence of cells styles, setting, 218 | worksheets, 87 |
| overriding the base currency, 67 | Query Designer button, 17 |
| 77 | query designer features, 23 |
| D. | 1 / 0 |
| P | D |
| Page dimension | R |
| searching for members in, 63 | Redo |
| passwords, changing | enabling in Smart View, 209, 210 |
| Essbase, 44 | using in Smart View, 210 |
| Paste Data Points button, 17 | Redo button, 17 |
| pasting cells in data forms, 64 | Refresh All button, 16 |
| percentage values, 64 | Refresh button, 16 |
| Pivot button, 16 | Refresh Offline Definition button, 18 |
| Pivot option, 108 | refreshing data |
| Point of View | while working offline, 203 |
| copying, 223 | refreshing documents, 166 |
| deleting, 224 | refreshing Reporting and Analysis documents, 166 |
| editing, 221 | Remove button, 18 |
| preserving formulas after POV changes, 216 | Remove Only button, 16 |
| selecting members for, 222 | Remove Only option, 110 |
| POV Manager, 221 | repeated members, suppressing, 208 |
| changing POV in Word or PowerPoint, 160 | replacement labels, 212 |
| POV Manager button, 17 | replacing #NoData label, 213 |
| POV toolbar | Report Designer, 51 |
| dragging and dropping members, 87 | chart, 53 |
| printing POV in header and footer, 224 | controlling reports, 53 |
| selecting members, 87 | creating reports, 51 |
| precedence, setting order in Cell Styles tab, 218 | deleting reporting objects, 54 |
| Preserve Formula on POV change option, 216 | function grid, 52 |
| preserving in free-form | keyboard equivalents, 338 |
| comments, 124 | slider, 53 |
| formats, 124 | table, 53 |
| formulas, 124 | Reporting and Analysis documents |
| Process Management grid display option, 213 | refreshing, 166 |
| Provider properties, 34 | updating, 166 |
| Trovider properties, or | reports |
| | asymmetric, 156 |
| Q | free-form, 119 |
| queries | resolving member names in free-form mode, 138 |
| creating, 87 | retaining members in spreadsheets, 211 |
| | O |

| retrieving | Smart View |
|---|---|
| Hybrid Analysis data, 108 | disabling, 28 |
| in free-form mode, 135 | enabling, 28 |
| increasing speed, 157 | Smart View toolbar |
| into asymmetric reports, 156 | displaying, 16 |
| performance impact, 157 | hiding, 16 |
| Rules on Form button, 17 | specifying |
| Rules on Form dialog box, 72 | latest time period, 102 |
| Run Reports button, 18 | UI colors, 214 |
| runtime prompts in business rules, 70 | specifying Dynamic Time Series members, 102 |
| rantime prompts in business rules, 70 | spreading data |
| | over time periods, 73 |
| S | overview, 74 |
| Scale grid display option, 215 | with cell locking, 77 |
| searching | Spreadsheet Add-in for Essbase |
| for members in the Page dimension, 63 | using concurrently with Smart View, 28 |
| Searching with Google, 18 | static link views. See dynamic link views |
| security certificate for Workspace servers, 168 | Subitems indentation option, 208 |
| Select Form button, 17 | Submit Data button, 16 |
| selecting | Submit Data option, 68 |
| folders and data forms to take offline, 200 | submitting data in ad hoc grids, 114 |
| members to take offline, 200 | Subset dialog box, 98 |
| Shared Services, adding data source through, 38 | subtotaling values, 67 |
| shortcut menus, 18 | supporting detail |
| Sibling Level zoom option, 210 | adding, 80 |
| silent installation, 28 | overview, 79 |
| Simulation and Forecasting Workbooks, 42 | totaling when cells are blank, 80 |
| adding and deleting workbooks, 43 | working with the hierarchy, 81 |
| managing workbooks, 43 | - |
| saving workbooks, 44 | Supporting Details button, 17 |
| setting properties, 43 | suppressing #Missing rows 208 |
| working with, 43 | #Missing rows, 208 |
| slider, 53 | invalid rows, 208 |
| creating from joined queries, 54 | No Access rows, 208 |
| creating from one query, 54 | No Data rows, 208 |
| Smart Lists, working with, 84 | repeated members, 208 |
| Smart Slices, 45 | types of data in spreadsheets, 207 |
| Creating queries, 46 | underscore character, 208 |
| using, 45 | zero value rows, 208 |
| smart tags | symmetric reports, 156 |
| displaying POV description using, 194 | Sync Back button, 17 |
| recognizing deleted, 196 | |
| removing a single instance, 195 | T |
| retrieving cell text using, 193 | table |
| retrieving entity currency using, 193 | Report Designer, 53 |
| retrieving functions, 198 | Take Offline button, 17 |
| retrieving single function value using, 192 | text |
| stop recognizing, 195 | adding in data forms, 65 |

A B C D E F G H I K L M N O P Q R S T U V W Z

| editing in data forms, 66 | Zoom In, 16 |
|---------------------------------------|---|
| entering free-form, 119 | Zoom Out, 16 |
| viewing in data forms, 66 | Totals indentation option, 208 |
| thousands separator in data forms, 63 | translating data in Ad Hoc grids, 113 |
| time periods, spreading data over, 73 | |
| time-related data | U |
| free-form mode, 150 | underscore character, suppressing, 208 |
| Query Designer, 92 | Undo |
| toolbar | enabling in Smart View, 209, 210 |
| displaying, 16 | using in Smart View, 210 |
| hiding, 16 | Undo button, 16 |
| toolbar buttons | uninstalling Smart View, 29 |
| Add, 18 | URL |
| Adjust, 17 | and data cells, 66 |
| BI Edit, 17 | launching from cell, 66 |
| Business Rules, 17 | linked to planning cells, 66 |
| Cell Text, 17 | using Smart View and Essbase Spreadsheet Add-in |
| Collapse, 17 | concurrently, 28 |
| Copy Data Points, 17 | |
| Data Source Manager, 16 | V |
| Expand, 17 | - |
| Function Builder, 17 | VBA functions, 231 |
| Instructions, 17 | HypCalculate, 241 |
| Keep Only, 16 | HypCall 242 |
| Lock, 17 Member Selection, 17 | HypCell, 243 |
| Member Selection, 17 | HypConnected 245 |
| Options, 18 Paste Data Points, 17 | HypConnected, 245 HypConnectionExists, 245 |
| Pivot, 16 | HypConnectToAPS, 246 |
| POV Manager, 17 | HypConsolidate, 247 |
| Query Designer, 17 | HypConsolidateAll, 247 |
| Redo, 17 | HypConsolidateAllWithData, 248 |
| Refresh, 16 | HypCreateConnection, 249 |
| Refresh All, 16 | HypCreateConnectionEX, 251 |
| Refresh Offline Definition, 18 | HypDeleteCalc, 252 |
| Remove, 18 | HypDisconnect, 253 |
| Remove Only, 16 | HypDisconnectEx, 254 |
| Rules on Forms, 17 | HypDisconnectFromAPS, 255 |
| Run Reports, 18 | HypDisplayToLinkView, 255 |
| Select Form, 17 | HypDisplayTwoDimArray, 256 |
| Submit Data, 16 | HypExecuteCalcScript, 256 |
| Supporting Details, 17 | HypExecuteMDXEx, 257 |
| Sync Back, 17 | HypExecuteQuery, 259 |
| Take Offline, 17 | HypFindMember, 260 |
| Undo, 16 | HypForceCalculate, 261 |
| Visualize in Excel, 18 | HypForceCalculateContribution, 262 |
| Visualize in HVE, 18 | HypForceTranslate, 263 |
| | 11/11 0100 11111011100, 200 |

| HypFreeDataPoint, 264 | HypRemoveOnly, 299 |
|--|---|
| HypGetAncestor, 264 | HypResetFriendlyName, 300 |
| HypGetChildren, 265 | HypRetrieve, 301 |
| HypGetColCount, 266 | HypRetrieveRange, 301 |
| HypGetColItems, 267 | HypSetActiveConnection POV, 302 |
| HypGetConnectionInfo, 268 | HypSetBackgroundPOV, 304 |
| HypGetCurrentAPSURL, 270 | HypSetCellsDirty, 304 |
| HypGetDataPoint, 270 | HypSetColItems, 305 |
| HypGetGlobalOption, 271 | HypSetConnAliasTable, 306 |
| HypGetLinkMacro, 273 | HypSetConnectionInfo, 307 |
| HypGetOverrideFlag, 274 | HypSetCurrentAPSURL, 308 |
| HypGetParent, 274 | HypSetGlobalOption, 309 |
| HypGetPOVCount, 275 | HypSetLinkMacro, 310 |
| HypGetPOVItems, 276 | HypSetMenu, 311 |
| HypGetRowCount, 277 | HypSetOverrideFlag, 312 |
| HypGetRowItems, 277 | HypSetPOV, 313 |
| HypGetSheetOption, 278 | HypSetPOVItems, 313 |
| HypGetSourceGrid, 280 | HypSetRowItems, 314 |
| HypGetSubstitutionVariable, 281 | HypSetSheetOption, 315 |
| HypIsAttribute, 283 | HypSetSubstitutionVariable, 317 |
| HypIsConnectedToAPS, 284 | HypSubmitData, 318 |
| HypIsDescendant, 284 | HypTranslate, 319 |
| HypIsExpense, 285 | HypUndo, 320 |
| HypIsParent, 286 | HypUseLinkMacro, 321 |
| HypIsUDA, 287 | HypZoomIn, 321 |
| HypKeepOnly, 288 | HypZoomOut, 323 |
| HypListCalcScripts, 289 | VBA functions, working with |
| HypMenuVCalculation, 324 | calling functions, 233 |
| HypMenuVConnect, 325 | creating a Visual Basic module, 231 |
| HypMenuVDisconnect, 325 | declaring functions, 232 |
| HypMenuVKeepOnly, 326 | dynamic link views, 234 |
| HypMenuVOptions, 326 | guidelines for declaring functions, 233 |
| HypMenuVPivot, 327 | menu functions, 323 |
| HypMenuVRefresh, 328 | migrating legacy VBA applications, 233 |
| HypMenuVRemoveOnly, 327 | parameters, 236 |
| HypMenuVRetrieve, 328 | return values, 236 |
| HypMenuVSubmitData, 329 | using functions in Smart View, 232 |
| HypMenuVZoomIn, 329 | Visualize in Excel button, 18 |
| HypMenuVZoomOut, 330 | Visualize in HVE button, 18 |
| HypMigrateConnectionToDataSourceMgr, 291 | |
| HypOtlGetMemberInfo, 291 | 14/ |
| HypPivot, 293 | W |
| HypPivotToGrid, 293 | Within Selected Group option, 157 |
| HypPivotToPOV, 294 | working with dynamic link views, 234 |
| HypQueryMembers, 295 | |
| HypRedo, 297 | Z |
| HypRemoveConnection, 298 | Zero value rows, suppressing, 208 |

Zoom In button, 16
Zoom in option, 104
Zoom options, 210
All Levels, 210
Bottom Level, 210
Next Level, 210
Sibling Level, 210
Zoom Out button, 16
Zoom out option, 105